(4)

ETL-0524 AD-A204 253

Consensus Theory in Expert Systems: An Adaptive Inference Framework and Application to Image Understanding

Kathryn B. Laskey
Marvin S. Cohen
William G. Roman
Paul K. Black
James R. McIntyre
Russell R. Vane, III

Decision Science Consortium, Inc.
Suite 300, 1895 Preston White Drive
Reston, Virginia 22091

620 0660

December 1988

Approved for public release; distribution is unlimited.

Prepared for:

U.S. Army Corps of Engineers
Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060-5546

89 2 3 112

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION: UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for Public Release; Distribution is Unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S): Technical Report 88-16 DSCI-TR-88-16	5. MONITORING ORGANIZATION REPORT NUMBER(S): ETL-0524
6a. NAME OF PERFORMING ORGANIZATION: Decision Science Consortium, In. — 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION: U.S. Army Engineer Topographic Laboratories
6c. ADDRESS (City, State, and ZIP Code): 1895 Preston White Drive, Suite 300 Reston, Virgina 22091	7b. ADDRESS (City, State, and ZIP Code): Fort Belvoir, Virginia 22060-5546
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: U.S. Army Engr Topographc Labs — 8b. OFFICE SYMBOL (If applicable): ETL	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: DACA72-86-C-0003
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)

Consensus Theory in Expert Systems: An Adaptive Inference Framework and Application to Image Understanding

12. PERSONAL AUTHOR(S)

Kathryn B. Laskey, Paul K. Black, Marvin S. Cohen, James R. McIntyre, Wm G. Roman, and Russell R. Vane,III

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final Technical	FROM 86 May TO 88 Dec	1988, December	57

16. SUPPLEMENTARY NOTATION

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)

Expert systems, belief functions, non-monotonic logic, assumption-based truth maintenance, image understanding

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Advances in automated image understanding technology are essential to our ability to exploit today's sophisticated imagery capabilities to support battlefield intelligence requirements. This report describes the application of a unique inference framework, Non-Monotonic Probabilist, to the problem of achieving consensus among modules, each of which supports a different part of the image understanding problem. Non-Monotonic Probabilist combines symbolic default reasoning with numerical uncertainty propagation to support a flexible ability to make and revise assumptions, to examine the degree of conflict associated with the current set of assumptions, and to resolve conflicts by reaching inside arguments and adjusting the underlying assumptions. Non-Monotonic Probabilist is a generic inference engine that is domain independent and can be applied to a variety of problems. Non-monotonic Probabilist has been embedded within COMMiTR, a consensus system intended to be incorporated within the Expert Resolution System at the U.S. Army Engineer Topographic Laboratories.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT: ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION: UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL: Joseph A. Rastatter	22b. TELEPHONE (Include Area Code): (202) 355-3856 — 22c. OFFICE SYMBOL

DD Form 1473, JUN 86 — Previous editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

PREFACE

This report describes work performed under contract DACA72-86-C-0003, entitled "Consensus Theory in Expert Systems" for the U.S. Army Engineer Topographic Laboratories, Fort Belvoir, Virginia, by Decision Science Consortium, Inc., Reston, Virginia. The Contracting Officer's Technical Representative was Mr. Joseph Rastatter.

CONTENTS

FIGURES

FIGURES

TABLES

1.0 SCOPE OF REPORT

This report describes the accomplishments of work performed for the U.S. Army Engineer Topographic Laboratories (ETL) under contract DACA72-86-C-0003, entitled "Consensus Theory in Expert Systems." The project described in this report is a Phase II Small Business Innovative Research project. During Phase I, DSC reviewed alternate theories for inference in expert systems, with the goal of identifying an approach, or combination of approaches, best suited to the problem of image understanding. Under that contract, the basic concept of the Non Monotonic Probabilist (NMP) was developed (Cohen et al, 1985). NMP reasoning uses a numerical (Shafer-Dempster) belief calculus, but embeds it within a qualitative reasoning framework. This allows reasoning at a meta-level, permitting the representation and manipulation of assumptions. This flexible capability to make and revise assumptions is essential for intelligent control of the application of an uncertainty calculus.

The goal of Phase II of this project was to implement NMP as a generic expert system building tool, and to demonstrate its use on a small scale expert system for understanding images. This report describes the accomplishment of this goal. Section 1 describes the purpose and organization of this report. Section 2 introduces the problem addressed by work under the present contract: integrating and achieving consensus among different automated image interpretation systems that operate at different levels and address different aspects of the interpretation problem. Section 3 describes the problem of obtaining timely and high-quality image intelligence on the battlefield, reviews work by ETL on different aspects of automated image interpretation, and describes the role of the present work as part of this endeavor. Section 4 describes the Consensus Module for Military Target Recognition (CoMMiTR) system developed under this contract, and the generic inference engine it uses. Section 5 describes how to use CoMMiTR. Section 6 summarizes the work performed under this contract and describes future directions the work might take.

2.0 INTRODUCTION

As the sophistication of image making technology continues to advance, it becomes increasingly necessary to develop image understanding technology that effectively exploits these advances. And indeed, the state of the art of automated image enhancement and understanding has been moving forward at a rapid rate. But a number of characteristics of imagery combine to create difficulties that continue to plague designers of image understanding systems.

The first difficulty is the sheer combinatorial complexity of images which may contain hundreds of millions of pixels of data. In part because of combinatorics, raw pixel data is often processed by low-level processing algorithms designed to perform extremely simple operations in parallel. Such exploitation of parallelism seems to be fundamental to humans' impressive image processing abilities. But human understanding also has an important cognitive dimension, which involves high-level reasoning that brings in knowledge about geometry and perspective, about the domain, about the situation in which the image was collected, and about the context within the image being analyzed. Artificial intelligence methods are well-suited to such symbolic, high-level reasoning, and the use of AI for image analysis has been increasing. Finally, uncertainty and incompleteness of information are ubiquitous in image understanding. Noise and clutter are always present in images, no matter how high the resolution; objects may be hidden because of other interfering objects or because of weather and lighting conditions; and the interpretation of part of an image is highly dependent on context which may be unknown at processing time. Probabilistic and related models seem then to have a place in image analysis.

Thus, image analysis has been characterized by application of a wide range of technologies. We have discussed three of the most common and most promising: low-level processing algorithms, symbolic artificial intelligence, and probabilistic modeling. We made a case for the importance of all three approaches, and we argue that each of these has its place in a unified approach to image understanding. Unfortunately, the approaches have not been well integrated--in part because of inherent difficulties of interfacing between different levels of analysis, and in part because adherents of the different approaches have tended to come from separate disciplines that communicate little with each other.

Typically, image processing occurs in several distinct stages. First, the image is enhanced by filtering out clutter and enhancing edges. Next, a low-level processing algorithm is applied to find regions and their boundaries. These two stages are computation intensive, and typically employ simple, repetitive algorithms highly suited to parallel or vector processors. The first stage commonly involves filtering techniques, which may be based on statistical models of error generation. The second stage often involves relaxation labeling (Rosenfeld et al., 1976). The final stage involves higher level reasoning about the objects found by the region growing algorithm. Expert systems have been employed for this stage. Usually these have been symbolic rule-based systems, but recently interest has been growing in applying frameworks for reasoning under uncertainty (e.g., Pearl, 1986; Laskey and Lehner, in press).

Usually these three levels operate entirely independently of each other. Yet it is generally accepted that the human ability to understand images stems from the ability to incorporate feedback between the different levels (cf., Rumelhart, et al., 1986). We suspect that an important contributing cause to the high rate of false alarms in existing low-level image processing systems is the inability to use feedback from higher-level processing to inform and direct low-level processing (Laskey, 1988). Laskey describes two approaches to providing such feedback. The first, deep integration, has the greatest theoretical coherence, but requires designing in feedback mechanisms at system development time. The second mechanism, surface integration, takes existing software packages and adjusts inputs and parameters in response to feedback from another processing level. Thus, surface integration may exploit existing image processing systems.

Section 4 of this report describes a prototype system, called CoMMiTR, developed by Decision Science Consortium to address the consensus problem. CoMMiTR was developed to perform surface integration of a number of systems comprising ETL's Expert Resolution System (described in Section 3 of this report). CoMMiTR simulates inputs from a number of different component systems, and uses global context to resolve conflicts to arrive at a consensus interpretation of the image. As a result of its conflict resolution procedure, CoMMiTR may recommend reanalysis by another module, and may suggest its own hypothesis as to the "correct" interpretation of that aspect of the image (e.g., map-to-image registration, labeling of objects, terrain type identification). Currently, the links necessary for other systems to process CoMMiTR's feedback have not been developed. However, CoMMiTR may be used to make suggestions to a human image analyst to check and possibly correct the output of the other modules.

CoMMiTR uses as an inference engine the Non-Monotonic Probabilist system (Cohen et al., 1985; Laskey, Cohen and Martin, 1988), described in Section 4.2. The Non-Monotonic Probabilist is a generic inference engine that combines a Shafer-Dempster belief calculus with a flexible ability to make and revise assumptions, to examine the degree of conflict associated with the current set assumptions, and to build in conflict resolution rules that automatically examine and resolve conflicts when they arise.

3.0 IMAGE UNDERSTANDING FOR BATTLEFIELD DECISION AIDING

3.1 USE OF IMAGERY IN THE MODERN BATTLEFIELD

The modern battlefield will increasingly require exploitation of sophisticated image processing technology. Image information is a major component of our intelligence gathering capability. Indeed, intelligence about the disposition of enemy forces is considered highly reliable only if it is confirmed by imagery. Significant advances in image technology have dramatically increased the amount of information that can be extracted by experienced analysts.

It is becoming increasingly apparent that the weak link in the chain from image data gathering to its final use in intelligence preparation of the battlefield is the process of analyzing the image and extracting the information relevant to conducting the battle. The sheer volume of image data on the future battlefield will be overwhelming, and its processing is highly time-critical. On top of this, enemy capability for deception is becoming increasingly sophisticated. Unfortunately, experienced image analysts are in short supply, and image processing is a time-consuming process. Thus, improvements in computer processing of imagery have become mandatory. Especially important is the ability to design processing systems that are capable of making judgments in the presence of uncertainty and incomplete information, while maintaining the ability to revise these judgments in the light of new information. The battlefield does not afford the luxury of waiting until all the evidence is in to take action.

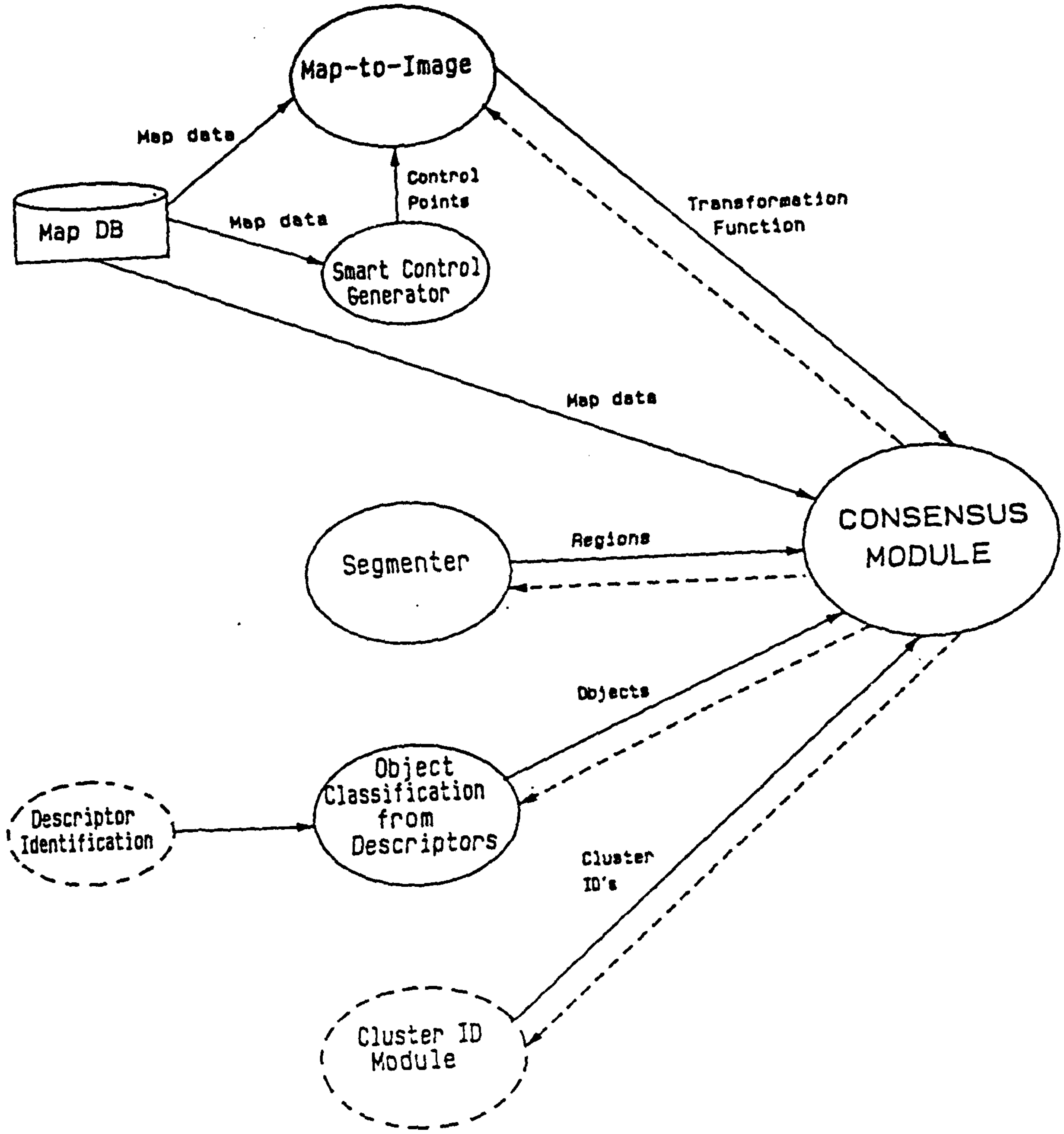
Image interpretation by trained human analysts makes extensive use of extra-image information. The image analyst is unlikely to have been fully briefed on the results of the IPB process, in part because it is desirable to have image information as independent as possible of other intelligence sources. Typically, though, the interpreter will be given an image and told to examine the image for specific information. For example, he or she will be asked to look for SAM sites in a certain vicinity, or to identify all maneuver units along the front. The fact that a certain kind of formation is expected to be found in the image is a powerful organizer of the interpreter's search. In addition, the analyst is usually versed in the information contained in OB books and handbooks about enemy capabilities, organization, and tactics. The analyst knows about the typical composition, disposition, frontages, depths, spacing, and signatures of enemy echelons and types of units for various capabilities and schemes of maneuver. The analyst can apply this information to the image, adjusting for modifications necessitated by terrain and weather conditions.

If automated image processing is to be successful, it must make use of the same kinds of information as informed human image interpreters. This requires the ability to do symbolic reasoning, and thus requires techniques from artificial intelligence. The most commonly applied paradigm within AI for reasoning about high-level knowledge such as force deployment patterns is the expert system. An expert system is built upon a set of rules specified in if-then form. (As an example, the system might have a rule stating that if a tank company is identified in a given area, then there should be a tank battalion in the vicinity.) Because the process of intelligence analysis is fraught with uncertainty, an expert system that reasons about order-of-battle requires some mechanism for processing uncertainty, such as probabilities or Shafer-Dempster beliefs. Cohen et al. (1985) provide a thorough review of alternate frameworks for reasoning under uncertainty, and suggests how they may be combined to exploit the best features of each. The work described herein is an implementation of the basic concept put forward by Cohen et al.

3.2 THE EXPERT RESOLUTION SYSTEM

The consensus system developed under this project was developed with the intent of fitting into the Expert Resolution System being developed at the U.S. Army Engineer Topographic Laboratories. This system involves combining a number of different kinds of analyses at different levels to arrive at a better overall interpretation of an image.

Figure 3-1 depicts the modules that make up the consensus system and their relationships. These modules are described below.



Arrows indicate flow of information.
Dotted lines indicate features not currently under development.

Figure 3-1. Expert resolution system.

3.2.1 Map Information

- *Map Databases.* The Expert Resolution System will incorporate on-line map databases, including Digital Terrain Elevation Data (DTED); Digital Feature Analysis Data (DFAD), and Vertical Obstruction Data (VOD).

- *Smart Control Generator.* The success of automated map-to-image registration depends heavily on the quality of the control features used in the registration algorithm. A smart control generator is being developed at ETL to perform this function. The smart control generator uses an expert system to generate control points from a MC&G database. Control points are objects in the MC&G database that are likely to appear prominently on an image, and thus serve as anchor points for a map-to-image registration module. The expert system generates control points using map data, sensor characteristics, and collection geometry. Sensor types include E/O, SAR, and IR. The control generator uses information about the sensor platform and its orientation with respect to the control feature to develop an initial estimate of the image coordinates of the control point. This initial estimate is then refined using the image itself (see below).

- *Map-to-Image Module.* A Map-to-Image module is being developed to transform image coordinates into map coordinates. The module takes as input a set of control points, together with their map coordinates and an initial estimate of their image coordinates. Control points are features that are expected to appear prominently on the image. A piecewise polynomial transformation function is generated as the registration proceeds through the image. Initial estimates of the image coordinates of the control points are updated during the registration process.

3.2.2 Image Information

- *Image Segmenter.* The Radar Image Classification Aid (RICA) uses a Bayes classifier to segment the image into the classification categories of city, field, forest, and water, and to determine the boundaries between regions of different classification. This function is performed from the image only, and does not use digitized map data.

- *Descriptors*. The Radar Descriptor System (RADES) classifies objects from the descriptor sets characterizing them. Descriptors are primitive features of the radar signature of the object (such as linear/curvilinear, or fine texture/medium texture). These descriptors form the input to RADES, which uses them to classify objects (areal, lineal, or special man-made) in the image. There is currently no module that identifies descriptor sets from the raw image; although input from such a module is required by RADES.

- *Military Target Cluster Identification*. The current version of the Expert Resolution system does not have a module that identifies military target clusters from raw image data. The RADES system does include identification of objects which could be classified as military targets (vehicles on road; aircraft), but its primary purpose is not to identify military targets. However, it is necessary for the consensus module to have input on tentative target classifications in order to correlate these with the other information it has. We therefore assume the availability of such a low-level classification system as one of the modules that provides input to the consensus module. A system such as the SAR Tactical Interpretation System (STIRS) could perform this role.

3.2.3 Consensus Module

The Consensus Module for Military Target Recognition (CoMMiTR) has the responsibility of correlating information from all the above sources, plus doctrinal information about enemy force dispositions, to *commit* to a consensus interpretation of the image. CoMMiTR receives from the target cluster identification module a list of tentative identifications of company-sized clusters. It attempts to group these into battalion formations that conform as closely as possible to doctrinal configurations. To do this, it needs to factor in terrain information such as whether there are rivers between units, or whether company sized clusters are at the same elevation, and cluster information such as type (e.g., maneuver company, artillery company) and posture (e.g., in convoy, field emplaced, moving to contact, in assembly area). Sometimes this information might lead to conflict. For example, a target cluster identification or target cluster grouping might conflict with a terrain feature identification (such as a river running between two companies in the same battalion, or a SAM site being found in the middle of a lake). When this happens, CoMMiTR invokes its conflict resolution mechanism, and calls into question the assumptions underlying the conflicting hypotheses. The system may then suggest that other systems reexamine their conclusions (i.e., it may suggest an error in the descriptor set classification of river; a misregistration of map to image; or a misidentification by the target cluster identification module).[1] Alternatively, when the conflict involves grouping of clusters, CoMMiTR may use the conflict to discredit the grouping relative to other possible groupings of the input clusters.

----

1. Currently, the interface links required for implementing retasking of other modules are not in place. In demonstrations of CoMMiTR, these links are simulated.

4.0 CoMMiTR: A CONSENSUS MODULE FOR MILITARY TARGET RECOGNITION

4.1 OVERVIEW

CoMMiTR has been designed by Decision Science Consortium to achieve consensus among different automated image interpretation systems in the particular case of military target recognition. CoMMiTR takes inputs from a number of different component systems comprising the Expert Resolution System developed at the U.S. Army Engineer Topographic Laboratories. The inputs have been simulated and stylized as the necessary links to the component systems have not been developed. However, CoMMiTR can provide information and suggestions that may cause a human image analyst to check the output of the low-level component systems. The system architecture of the consensus module used by CoMMiTR is depicted in Figure 4.1.

CoMMiTR performs a *surface integration* of the component low-level systems; simulating inputs from these systems to arrive at a consensus interpretation of the image and, in the case of conflict, recommending re-analysis by the component systems. The inputs for CoMMiTR are taken from a number of different component systems (see Section 3). These include DTED (Digital Terrain Elevation Data), and DFAD (Digital Feature Analysis Data). CoMMiTR also simulates inputs from existing image analysis programs such as RICA (Radar Image Classification Aid) and RADES (Radar Descriptor System). These inputs are simulated because a direct link between these systems and CoMMiTR has not been developed. CoMMiTR also simulates inputs from a system such as STIRS (SAR Tactical Interpretation System) that may provide identification of objects which could be *classified as military targets*.

CoMMiTR correlates the information from the above sources and incorporates doctrinal information about the disposition of enemy forces to reach a consensus interpretation of the image. The image is essentially a simulated SAR image occupying a 10 x 30k area which is 5k behind enemy lines. The actual simulated inputs consist of a list of identifications of company-sized clusters (see Section 5). CoMMiTR attempts to group these companies into battalion formations that conform as closely as possible to doctrinal configurations of enemy forces. Information about location with respect to the FEBA,

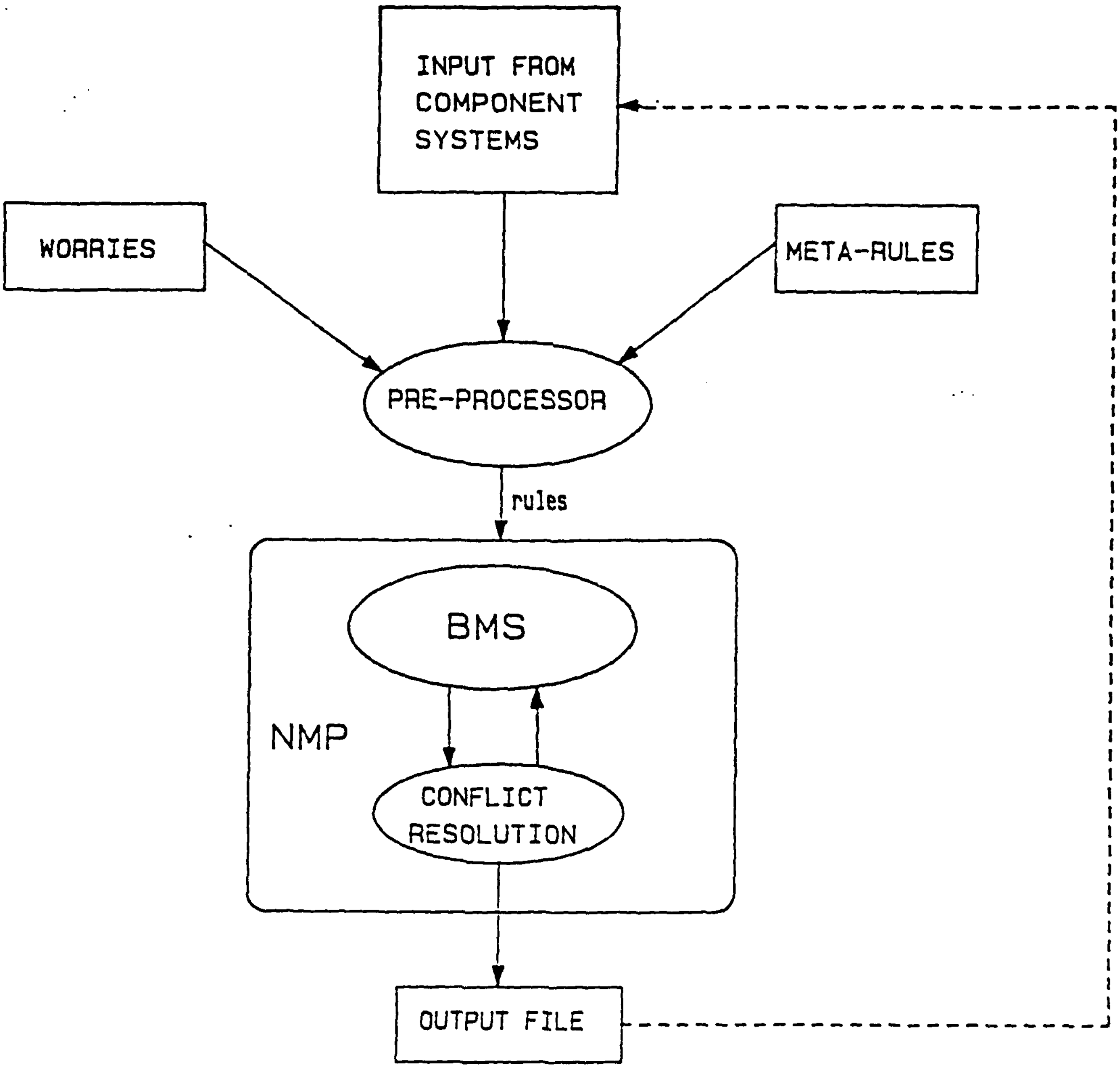


Figure 4-1. Consensus module.

terrain features and map-to-image registration are factored in to achieve a consensus interpretation of the image.

For instance, four companies may be located in such a way that they form a battalion. However, if a river runs between these companies, or if there is a mountain between the companies, CoMMiTR is less likely to conclude that particular battalion formation (unless a bridge exists across the river, or a tunnel through the mountain, etc.). CoMMiTR reasons about these obstacles, e.g., rivers, mountains, bridges, etc., by forming default rules or assumptions (e.g., assume there is *not* a river between two companies unless there is evidence to the contrary). When evidence conflicts with the assumptions, CoMMiTR's conflict resolution mechanism is invoked. At this point, CoMMiTR may suggest reanalysis within one or more of the component low-level systems (e.g., if a company is found in the middle of a lake, CoMMiTR may suggest reanalysis of the map-to-image registration module).

The rules participating in this example may be paraphrased as:

```
R11: (IF (Companies conform-to Maneuver-Battalion-doctrine)
         THEN (ID Maneuver-Battalion))
```

which states that if a group of company sized clusters conform to doctrinal requirements for enemy maneuver battalions, then they are believed to form a maneuver battalion. Rules in CoMMiTR also depend on a background context. For example the above rule depends on terrain features, distance from the FEBA, and map to image registration. Typical rules are better represented by the following generic example:

```
R1:  (IF <antecedents> THEN <consequents>
                            PROVIDED <background antecedents>).
```

such that the above example may be written as:

```
R11: (IF (Companies conform-to Maneuver-Battalion-doctrine)
         THEN (ID Maneuver-Battalion)
         PROVIDED (No-river between-companies))
```

If two or more rules lead to conflicting conclusions then the conflict resolution mechanism is invoked to identify which background antecedent assumptions may need to be reexamined and perhaps revised. Conflict resolution may have the effect of altering the belief in the background assumptions. Rules are described more fully in Section 4.2.

The above example described a situation where the conflict is linked directly to the input from the component low-level systems. Other forms of conflict can arise within CoMMiTR. For example, when companies are near the edge of the image sector, the remaining companies that may form a battalion could be out of range. This may cause the system to generate conflict, because of a default rule that all units must belong to some higher level structure.

Furthermore, once battalions have been formed from companies, they could be processed in a similar fashion to determine regimental formations. This feature could easily be added to CoMMiTR with the inclusion of the relevant set of rules. This could also create a further possibility for conflict: If the regiment interpretation is not progressing smoothly, CoMMiTR could then reevaluate at the battalion level.

The Shafer-Dempster belief calculus is used to represent uncertainty about the rules in CoMMiTR. The reasons for using Shafer-Dempster beliefs are more fully explained in Section 4.2. Its advantages are its ability to represent *evidential incompleteness* and its capability for naturally expressing conflict between hypotheses. This calculus is used within the *Non-Monotonic Probabilist Inference Engine* (see Section 4.2) along with a process of *default reasoning* which is applied directly to the beliefs themselves. Default reasoning consists of making assumptions; in cases where conflicting hypotheses are competing, these assumptions may be revised as part of the conflict resolution mechanism.

The Non-Monotonic Probabilist uses the *Belief Maintenance System* (Laskey and Lehner, 1988) to compute beliefs, keep account of assumptions and compute the degree of conflict between competing hypotheses. The Belief Maintenance System is a calculating device. Higher level control of reasoning is done by the *Evidential Reasoner* which passes information to the BMS. The BMS processes this information and passes information back to the Evidential Reasoner. These operations are fully described in Section 4.2.

User inputs for CoMMiTR consist of a scenario and a "worry." Image interpretation by human analysts typically involves searching for specific information, which we have denoted as a "worry". For example, the analyst may be asked to look for all maneuver battalions in convoy. CoMMiTR provides this capability in the form of a "worry list" which directs CoMMiTR's processing.

The scenarios depict centers of location of company sized elements, together with company-type (e.g., maneuver, artillery, SAM, etc.) and company-posture (e.g., in convoy, in assembly area, moving to contact, field emplaced) information. A scenario generator can be used to input scenarios in the form (x-coordinate, y-coordinate, company-type, company-posture). Examples of scenarios currently implemented for CoMMiTR are given in Appendix II. Information concerning terrain and map features such as doctrinal information, and positions of rivers or ridges with respect to company-size elements are input directly as rule premises (e.g., there is a river between company-1 and company-2; company-1 is near the edge of the image sector; etc.). These premises may have associated degrees of belief, or may be assumed by default to have belief 1. Default beliefs may be revised by CoMMiTR when they no longer seem appropriate. A detailed example is given in Section 4.3.

Rule construction is also described in Section 4.3. Rules in CoMMiTR are constructed from "meta-rules" and a list of premises. The meta-rules are specified for each Battalion type and posture. The premise list consists of doctrinal information for enemy battalion formations. Once the rules have been established, further processing generates a set of "solution" rules. These ultimately provide the output from CoMMiTR.

Once the pre-processor has generated the rule-set (including premises and solution rules), the rules are passed to the NMP (see Section 4.2). The BMS then calculates beliefs for the solution rule set. The conflict resolution mechanism is used to resolve conflicts and generate a consensus. This process is explained in detail with an example in Section 4.2.

The final output consists of a file that can be read by the local editor. The output file consists of the most likely battalion formations given the companies comprising the scenario. The output is also specific to the particular "worry" selected.

4.2 NON-MONOTONIC PROBABILIST INFERENCE ENGINE

The non-monotonic probabilist (NMP) reasons with numerical degrees of belief, but in addition can represent the degree of shiftability of its own arguments in response to unexpected or conflicting evidence. NMP was first proposed by Cohen (1985), and is outlined in greater detail by Laskey, Cohen and Martin (1988). This section summarizes how NMP works in the context of a simplified image understanding example. Domain knowledge in NMP is structured around a schema for representing an evidential argument. The argument schema makes explicit the background context within which an inference rule is valid, enabling the system to call into question and revise its background assumptions when they no longer appear to be valid.

NMP arguments are combined and chained together using a Shafer-Dempster belief calculus embedded within a process of default reasoning applied to the beliefs themselves. Nonindependencies due to shared premises are automatically accounted for in the belief calculations. Default reasoning serves to control the application of the belief calculus. Its role is to keep track of assumptions and to direct the process of belief revision when those assumptions lead to anomalous results.

4.2.1 NMP Argument Structure

Arguments in NMP are represented by an argument schema based on the one developed by Toulmin et al. (1984). In Toulmin's schema (Figure 4-2), a claim, or conclusion whose merits we are seeking to establish, is supported by grounds, or evidence. The basis of this support is the existence of a warrant that states the general connection between grounds and claim. The warrant might for example be a general rule that this type of ground provides basis for this type of claim. The backing provides an explanation of why the warrant is regarded as reliable. That is, it provides theoretical or empirical evidence

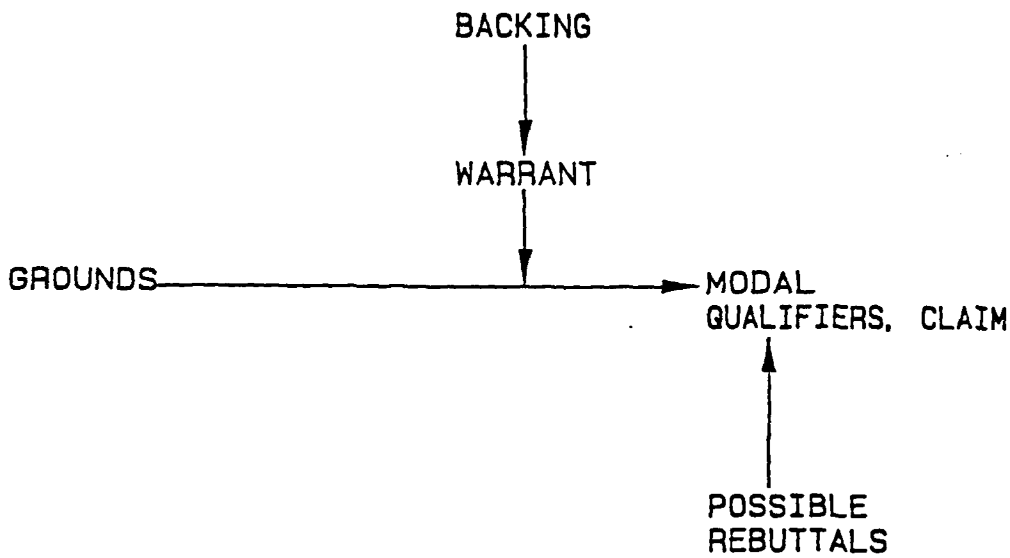


Figure 4-2. Toulmin's structure for an evidential argument.

for the existence of an evidential relation or causal connection between grounds and claim. Modal qualifiers (e.g., probably; possibly) weaken or strengthen the validity of the claim. Possible rebuttals deactivate the link between grounds and claim by asserting conditions under which the warrant is invalid. A way of reading this structure is: Grounds, so Qualified Claim, unless Rebuttal, since Warrant, on account of Backing.

Figure 4-3 shows how Toulmin's argument schema has been applied in the context of NMP. An argument from evidence to a conclusion is constructed using as a warrant a rule asserting that an evidential link exists between them. This rule may in turn be backed by a deeper theoretical or causal model, such as a general law or a statistical analysis. The evidential argument may be invalidated if any assumptions underlying the model do not hold.

4.2.2 The Belief Calculus

NMP arguments are combined and chained together using a Shafer-Dempster belief calculus embedded within a process of default reasoning applied to the beliefs themselves. Shafer's theory was chosen for our implementation because of several features that make it amenable to an intelligent control and belief revision capability:

- *Representing evidential incompleteness.* Usually in military intelligence problems our evidence is incomplete. According to Shafer (Shafer and Tversky, 1985), the contrast between belief functions and probabilities focuses directly on this idea of incompleteness of evidence. While the probability of a hypothesis measures the chance that it is true conditional on given evidence, its Shafer-Dempster belief measures the degree to which the evidence *means* (or proves) that it is true (see also Pearl, 1988, chapter 9). By stressing the link between evidence and hypothesis, Shafer's theory is able to provide an explicit measure of the quality of evidence or degree of ignorance.

- *Diagnosis of conflict.* To the extent that two arguments support incompatible hypotheses, combining beliefs by Dempster's Rule creates support

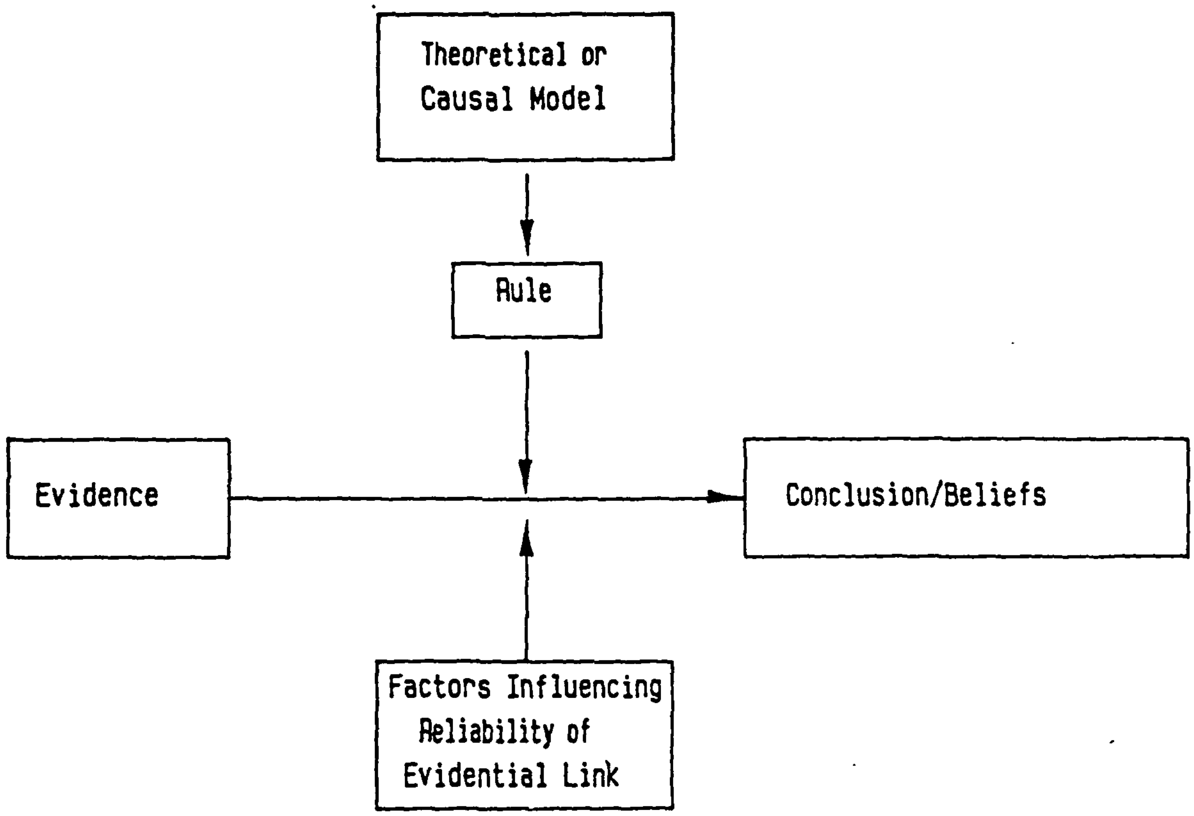


Figure 4-3. Argument structure in NMP.

for the null set. This support is then removed by proportionately increasing support for all non-null sets. But null set support serves a useful function for NMP. It measures the degree to which propositions are inconsistent, and thus constitutes a natural measure of conflict in the evidence.

- *Assumptions.* To the degree that current evidence is uncommitted with regard to the truth or falsity of a hypothesis, there is room for assumptions. An assumption could be naturally represented in Shafer's framework as a decision regarding the allocation of uncommitted belief. Such a decision, by definition, goes beyond the evidence, but remains within the constraints of the evidence.

- *Discrediting arguments.* The outcome of a process of conflict resolution may be the discrediting of one or more lines of reasoning that led to the conflict, by rejecting assumptions involved in those arguments. Partial or complete rejection of an assumption is represented by decreasing, possibly to zero, the degree to which uncommitted belief is allocated to the formerly assumed hypothesis.

Shafer himself does not address the notion of an assumption, as just outlined. Indeed, actions in response to conflict, such as re-examining source credibility, must occur outside the theoretical structure of belief functions. Non-monotonic probabilist embeds a belief function model within a qualitative assumption-based reasoning process. This qualitative reasoning process uses the tools implicit within Shafer's calculus to formalize and direct an iterative conflict resolution and assumption revision process.

4.2.3 Belief Propagation

Non-Monotonic Probabilist uses the belief maintenance system (BMS) (Laskey and Lehner, 1988) to compute beliefs and keep track of assumptions. The BMS represents belief functions as tokens attached to rules linking evidence and conclusions. Stored with each token is probabilistic information about the strength of the evidential link it represents. A probability calculus on belief tokens is formally equivalent to a Shafer-Dempster calculus (Laskey and

Lehner, 1988; in press). An explicit provision for making and revising assumptions has been added to complete the machinery necessary for implementing NMP.

Belief maintenance combines deKleer's (1986a,b,c) assumption-based truth maintenance system (ATMS) with a module for representing and reasoning with degrees of belief on symbolic tokens manipulated by the ATMS. DeKleer argues for explicit separation of reasoning into two functions, problem solving and truth maintenance. In NMP, the *belief* maintenance system performs a role analogous to the role deKleer proposes for *truth* maintenance. The BMS keeps account of assumptions, computes beliefs, determines the degree of conflict, and attributes that conflict to specific assumptions. It therefore supports a set of basic functions necessary for NMP's adaptive control and conflict resolution. NMP's analogy to deKleer's problem solver is the *evidential reasoner*: a higher level system that encodes argument schemas and information justifying applications of the rules. The evidential reasoner constructs arguments symbolically, and passes to the BMS the task of computing beliefs and conflict. The BMS does not "understand" the beliefs it computes--it treats its tokens as uninterpreted symbols with belief numbers attached to them. The evidential reasoner uses the output of the BMS (beliefs and conflict) to control application of the rules and direct conflict resolution.

Two features of the ATMS make it well-suited to its role as the substrate for belief maintenance. First, it is designed to be able to maintain belief simultaneously for multiple and possibly inconsistent propositions, a capability required for reasoning with numerical beliefs. Second, the design of the ATMS maintains an explicit separation between problem solving and truth maintenance. In our terms, this means that high-level reasoning about the application of the inference mechanism is explicitly separated from (although informed by) the process of keeping track of assumptions and computing beliefs. Belief maintenance is capable of representing the full generality of the Shafer-Dempster calculus. The ATMS automatically keeps account, in symbolic form, of the propagation of beliefs through chains of inference, nonindependencies created through shared premises, and inconsistent combinations of tokens. The belief computation module incorporates all this information to compute correct Shafer-Dempster beliefs when requested. Adding to this

framework the capability to make and reason with default assumptions results in a fully integrated symbolic and numeric uncertainty management framework. This framework is well suited to qualitative reasoning about the application of a numeric uncertainty calculus.

A general formal presentation of how assumption-based truth maintenance can be used to encode and reason with belief functions is given in Laskey and Lehner (in press). In Laskey and Lehner (1988), this framework is extended to allow making and revising default assumptions.

An NMP rule has the general form:

```
(IF <antecedents> THEN <consequent>
                  PROVIDED <background antecedents>) .
```

Typically, the effect of the background context is summarized by a numerical belief value, representing the degree to which the evidence is taken to imply the conclusion. Thus, the system might have the rule:

```
R38: (IF (Template_Matcher_Report Cluster Maneuver_Company))
      THEN (ID Cluster Maneuver_Company)
      PROVIDED (RULE-VALID-R38)),
```

which states that if the template matcher identifies a cluster as a maneuver company, then it is believed to be a maneuver company, provided RULE-VALID-R38. The symbol RULE-VALID-R38 represents a *belief token*, a special construct within the BMS that carries an attached probability. For example, if the assigned probability is .8, then a the report will cause the ID of the cluster to be assigned .8 belief in Maneuver_Company, absent other evidence. The probability of the belief token RULE-VALID-R38 may be interpreted as the probability that the rule is "working". That is, this probability summarizes our belief that some condition disabling the rule has not occurred.

The ATMS propagates tokens, including belief tokens, through chains of argument. It maintains a *label* for each proposition in its database, which represents the token sets that are sufficient to prove the proposition. In the above example, after receiving the report (Template_Matcher_Report Cluster

Maneuver_Company) the label of (ID cluster Maneuver-Company) would be:

(ID Cluster Maneuver_Company): {RULE-VALID-R38} .

The ATMS can chain arguments together, and form multiple arguments for the same conclusion. For example, suppose we also had the label:

(In_Maneuver_Battalion Cluster): {RULE-VALID-R19},

as the result of firing another rule. Firing the rule

```
R30: (IF (In_Maneuver_Battalion Cluster) THEN (ID Cluster Maneuver_Company)
                                         PROVIDED (RULE-VALID-R30))
```

changes the label of (ID cluster Maneuver_Company) to:

```
(ID cluster Maneuver_Company):
     {RULE-VALID-R38}, {RULE-VALID-R19,RULE-VALID-R30} .
```

This means that the cluster can be proven *to* be *a* maneuver company if RULE-VALID-R38 is true (i.e., Rule 38 is "working"), or if RULE-VALID-R19 *and* RULE-VALID-R30 are both true.

The probability of a proposition's label is the probability that the proposition can be proven--that is, its Shafer-Dempster belief. In our example, to find the degree of belief in (ID Cluster Maneuver-Company), we need to find the probability of (RULE-VALID-R38 *or* (RULE-VALID-R18 *and* RULE-VALID-R30)). To do this, the probability calculator module of the BMS constructs a "truth table" representing all possible truth values of the belief tokens in the proposition's label. The probability of the label is then the probability of the rows in the truth table that imply the label (i.e. in which RULE-VALID-R38 is true or RULE-VALID-R19 and RULE-VALID-R30 are both true). Figure 4-4 shows how this is done for this example, assuming beliefs .8, .7, and .9 for RULE-VALID-R38, RULE-VALID-R19, and RULE-VALID-R30, respectively.

RULE-VALID-R38	RULE-VALID-R19	RULE-VALID-R30	Conclusion	Belief
T	T	T	T	.50
T	T	F	T	.06
T	F	T	T	.13
T	F	F	T	.01
F	T	T	T	.22
F	T	F	?	.02
F	F	T	?	.05
F	F	F	?	.01

RULE-VALID-R38: P(T) = .7; P(F) = .3
RULE-VALID-R19: P(T) = .8; P(F) = .2
RULE-VALID-R30: P(T) = .9; P(F) = .1

Belief in (ID Cluster Maneuver_Company) = .92

Figure 4-4. Truth table for (ID Cluster Maneuver_Company).

4.2.4 Assumptions in NMP

As noted above, the ability to make and revise assumptions was an important design criterion for NMP. Often we wish the system to assume a high belief for a rule unless there is direct evidence to the contrary, even if this high belief is not directly justified by the evidence.

For example, consider the rule:

```
R39: (IF ((Template_Matcher_Report Cluster SAM)) THEN (ID Cluster SAM)
                                                  PROVIDED (RULE-VALID-R39)).
```

In fact, on the basis of a template match alone, we might not be justified in concluding the existence of a SAM site. We might be justified only in concluding that there is either a SAM or a decoy, replacing the above rule with:

```
R39': (IF ((Template_Matcher_Report Cluster SAM)) THEN (or (ID Cluster SAM)
                                                               (ID Cluster decoy))
                                                  PROVIDED (RULE-VALID-R39)).
```

NMP allows the system to make an *assumption* which has the effect of strengthening the latter rule to operate like the former rule. But the system keeps track of the assumption it made, and can revise the assumption if it is found to conflict with other evidence.

Assumptions are implemented within NMP by assigning *default tokens*, or special tokens that are treated as if they had probability 1. To make such an assumption in NMP, we would use two rules. First would be Rule 39' above, which concludes on the basis of a template match that either a SAM or a decoy is present. Second, the system would encode the following rule:

```
R40: (IF ((Template_Matcher_Report Cluster SAM))
     THEN (ID Cluster SAM)
     PROVIDED (RULE-VALID-R39 ASSUME-R40)) .
```

The token ASSUME-R40 is a *default token*, which is treated as if it has probability 1 until the system encounters evidence that makes it question its original assumption.

Thus, Rule 39' left belief *uncommitted* between (ID Cluster SAM) and (ID Cluster decoy), assigning the belief to their disjunction rather than to either individually. What Rule 40 does is to allocate this uncommitted belief *by default* to the more specific hypothesis (ID Cluster SAM). Thus, when a template matcher report of SAM comes in, the cluster ID labels become:

(ID Cluster SAM): {RULE-VALID-R39,ASSUME-R40}.
(or (ID Cluster Sam) (ID Cluster decoy)): {RULE-VALID-R39}

As we have said, default tokens are treated by the probability calculator as if they had probability 1. Thus, in this example, the belief in (ID Cluster SAM) is equal to the probability of the belief token RULE-VALID-R39 (assuming the above rules are the only evidence related to the ID of this particular cluster).

In the above rule, we assumed that *all* of the belief assigned to (or (ID Cluster SAM) was to be allocated to the stronger hypothesis (ID Cluster decoy)). We might wish to allocate only *part* of this belief, leaving some of it uncommitted (that is, not distinguishing between a SAM or a decoy). This could be done by replacing the allocation rule with:

```
R40': (IF ((Template_Matcher_Report Cluster SAM))
       THEN (ID Cluster SAM)
       PROVIDED (RULE-VALID-R39 RULE-VALID-R40 ASSUME-R40)) .
```

Now the labels become:

(ID Cluster SAM): {RULE-VALID-R39,RULE-VALID-R40,ASSUME-R40}
(or (ID Cluster Sam) (ID Cluster decoy)): {RULE-VALID-R39}

As the truth table in Figure 4-5 demonstrates, the effect of this rule is to allocate a *percentage* of the uncommitted belief to (ID Cluster SAM), this percentage being equal to the probability of the belief token RULE-VALID-R40 (60% in this case). (Note that the assumption ASSUME-R40 need not explicitly appear in the truth table, since it is assumed to have truth value T for all rows.)

RULE-VALID-R39	RULE-VALID-R40	(ID Cluster SAM)	(or (ID Cluster SAM) (ID Cluster decoy))	Belief
T	T	T	T	.54
T	F	?	T	.36
F	T	?	?	.06
F	F	?	?	.04

RULE-VALID-R39: P(T) = .9; P(F) = .1
RULE-VALID-R40: P(T) = .6; P(F) = .4

Belief in (ID Cluster SAM) = .54
Belief in (or (ID Cluster SAM) (ID Cluster decoy)) = .36

Figure 4-5. Truth table for belief allocation to SAM.

4.2.5 Conflict Resolution in NMP

Representing assumptions explicitly is useful because the system can examine them and revise them when necessary. In NMP, assumptions may be revised in response to *conflict*. Conflict occurs when arguments support contradictory conclusions.

Let us consider an example. Suppose the system had a default rule stating the system's belief that no SAM emissions are emanating from an area if there is no specific evidence of emissions:

```
R03: (IF () THEN (not (SAM_Emissions_Near (Loc Cluster)))
           PROVIDED (ASSUME-R03))
```

This produces the label:

```
(not (SAM_Emissions_Near (Loc Cluster))):  {ASSUME-R03}
```

Now suppose we have another rule:

```
R25: (IF ((ID Cluster SAM)) THEN (SAM_Emissions_Near (Loc Cluster))
                                 PROVIDED (RULE-VALID-R25 ASSUME-R25))
```

After firing R40' as described in Section 3.2.4, firing this rule results in the label:

```
(SAM_Emissions_Near (Loc Cluster)):
    {RULE-VALID-R39,RULE-VALID-R40,RULE-VALID-R25,ASSUME-R40,ASSUME-R25}
```

Because the system knows that (SAM_Emissions_Near (Loc Cluster)) and its negation are inconsistent, it creates a *nogood* environment by combining their labels:

nogood {RULE-VALID-R39, RULE-VALID-R40, RULE-VALID-R25, ASSUME-R40, ASSUME-R25, ASSUME-R03}.

Nogood environments are sets of assumptions that cannot all be true (note that if all the above tokens were true we could derive both (SAM_Emissions_Near

(Loc Cluster)) and its negation). Figure 4-6 illustrates the belief computations for this example.

Note the high degree of belief assigned to inconsistent sets, or the contradiction ⊥. Rows of the truth table are marked contradictory if, coupled with the current defaults, they are *nogood*. The degree of belief assigned to ⊥ by the belief calculator algorithm is the *conflict* associated with the hypotheses (SAM_Emissions_Near (Loc Cluster)) and (not (SAM_Emissions_Near (Loc Cluster)). When this number gets large, the system examines the assumptions contributing to the conflict for possible revision.

In our example, revising any of the three assumptions (ASSUME-R40, ASSUME-R25, or ASSUME-R03) would remove the conflict. The final beliefs the system is left with, however, depends critically on which is revised. Dropping either of the assumptions ASSUME-R40 or ASSUME-R25 would disrupt the chain of evidence leading to the conclusion (SAM_Emissions_Near (Loc Cluster)), setting its belief to zero, with belief in its negation remaining at 1.0. Removing the assumption ASSUME-R03 removes the argument for (not (SAM_Emissions_Near (Loc Cluster)), leaving its belief equal to zero. Belief in (SAM_Emissions_Near (Loc Cluster)) is then given by the analysis in Figure 4-7.

4.2.6 Modes of Conflict Resolution in NMP

The system designer can write conflict rules into NMP, instructing it how to deal with conflict arising from application of its rules. Each conflict rule instructs NMP how to handle conflicts between certain rules or classes of rules within NMP.

Recall that Rule 03 in the example of Section 3.2.5 was a default rule that was supposed to apply only when there was no direct evidence of SAM emissions. If the existence of a SAM is to be taken to constitute direct evidence of SAM emissions, then the conflict rule would instruct the system that ASSUME-R25 is to take precedence over ASSUME-R03. If, on the other hand, only observation of emissions will suffice, then the conflict rule would state that ASSUME-R03 takes precedence over ASSUME-R40. Alternately, ASSUME-R40 may take precedence over ASSUME-R03, but ASSUME-R03 may take precedence over ASSUME-R25 (meaning

RULE-VALID-R39	RULE-VALID-R40	RULE-VALID-R25	(SAM-Emissions _Near (Loc Cluster))	Belief
T	T	T	⊥	.43
T	T	F	F	.11
T	F	T	F	.29
T	F	F	F	.07
F	T	T	F	.05
F	T	F	F	.01
F	F	T	F	.03
F	F	F	F	.01

RULE-VALID-R39: P(T) = .9; P(F) = .1
RULE-VALID-R40: P(T) = .6; P(F) = .4
RULE-VALID-R25: P(T) = .8; P(F) = .2

Belief in (SAM_Emissions_Near (Loc Cluster)) = 0
Belief in (not (SAM_Emissions_Near (Loc Cluster)) = .57
Conflict = .43

Figure 4-6. <u>Example of beliefs when evidence conflicts</u>.

RULE-VALID-R39	RULE-VALID-R40	RULE-VALID-R25	(SAM-Emissions _Near (Loc Cluster))	Belief
T	T	T	⊥	.43
T	T	F	?	.11
T	F	T	?	.29
T	F	F	?	.07
F	T	T	?	.05
F	T	F	?	.01
F	F	T	?	.03
F	F	F	?	.01

RULE-VALID-R39: P(T) = .9; P(F) = .1
RULE-VALID-R40: P(T) = .6; P(F) = .4
RULE-VALID-R25: P(T) = .8; P(F) = .2

Belief in (SAM_Emissions_Near (Loc Cluster)) = 0
Belief in (not (SAM_Emissions_Near (Loc Cluster))) = 0
Conflict = 0

Figure 4-7. Belief after conflict resolution (ASSUME-R03 dropped).

that the lack of emissions casts doubt on our assumption that the SAM was not a decoy). Sometimes, conflict rules do not specify which assumptions are to be dropped. This may mean that the conflict cannot be removed given current information. In this case, the conflict rules may recommend information gathering to resolve the conflict.

Conflict rules may also invoke the user to arbitrate conflicts. In this case, the user may analyze the context in which the conflict occurs, examine beliefs of relevant hypotheses, trace relevant arguments, and make a decision about which assumptions should be revised.

4.3 APPLYING NMP TO THE CONSENSUS PROBLEM

Inputs to CoMMiTR consist of a scenario and a set of "worries" (see Section 5). The scenario consists of a set of centers of location of company sized elements along with company type (e.g., maneuver, artillery, SAM, etc.) and company-posture (e.g., in convoy, in assembly area, moving to contact, field emplaced). Worries enable the user to focus the system on interpreting particular company-types and company-postures. For example, a typical "worry" could be to determine all the maneuver battalions that are currently in assembly area.

To demonstrate our implementation of NMP within CoMMiTR, an example involving maneuver battalions in assembly area is followed through the succeeding subsection (i.e., the company type is 'Maneuver' and the company posture is 'in assembly area').

According to doctrine about enemy forces disposition, maneuver battalions in assembly area are most likely to consist of four companies, but possibly three or five. The centers of these companies doctrinally form a regular polygon (with appropriate number of sides) with doctrinal distance between centroids in the range 500 to 800 meters (depending on number of companies and terrain features). This example is now followed through the stages of rule formation, belief computation and conflict resolution to a final solution.

The first example has four companies which are labeled A1 through A4 (see

Figure 4-8). The doctrinal angle between the centers of three adjacent companies is 90 degrees; the doctrinal distance is between 500 and 800 meters.

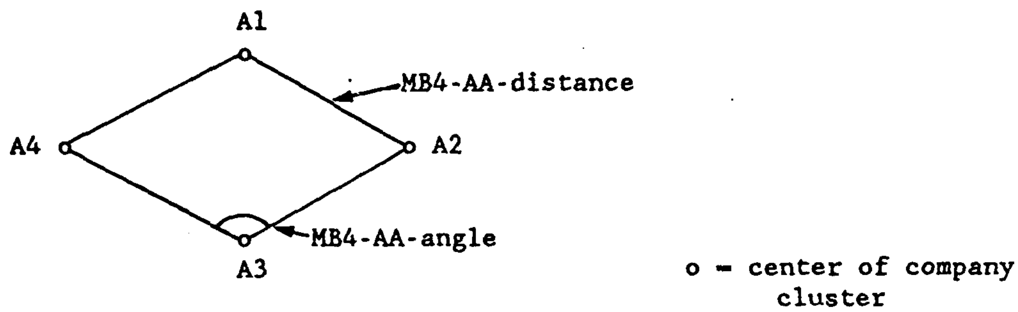


Figure 4-8. Four maneuver companies in assembly area.

Clearly, these distances and angles are variable and some leeway is allowed before completely discounting belief in a battalion formation. A set of four maneuver companies in assembly area are doctrinally defined by four distances (A1-A2, A2-A3, A3-A4, A4-A1) and two adjacent angles (e.g., A2-A1-A4, A1-A4-A3).

The first step in the pre-processing is to form a list of premises for the four companies. These premises concern the doctrinal distances between companies, and the doctrine that companies are spaced evenly (this is implemented by comparing the angles between companies with the angles that would be observed if they were spaced evenly). The premise list consists of all the combinations of companies that satisfy the doctrinal requirement. For example, suppose A1 and A3 do not meet the doctrinal requirement for distance, and that this is the only exception in the case of distance. The premise list then looks like this:

```
(((MB4-AA-distance A1 A2) x1)
 ((MB4-AA-distance A1 A4) x2)
 ((MB4-AA-distance A2 A1) x3)
 ((MB4-AA-distance A2 A3) x4)
 ((MB4-AA-distance A2 A4) x5)
 ((MB4-AA-distance A3 A2) x6)
 ((MB4-AA-distance A3 A4) x7)
 ((MB4-AA-distance A4 A1) x8)
 ((MB4-AA-distance A4 A2) x9)
 ((MB4-AA-distance A4 A3) x10)
```

```
((MB4-AA-angle A1 A2 A3) x11)
((MB4-AA-angle A1 A4 A3) x12)
((MB4-AA-angle A2 A1 A4) x13)
((MB4-AA-angle A2 A3 A4) x14)
((MB4-AA-angle A3 A2 A1) x15)
((MB4-AA-angle A3 A4 A1) x16)
((MB4-AA-angle A4 A1 A2) x17)
((MB4-AA-angle A4 A3 A2) x18))
```

The list of premises contains all cases where the doctrinal requirements are satisfied to some degree (other than 0). The x_i represent that degree and may be thought of as normalized distances and angles ($0 < x_i < 1$).

CoMMiTR also contains a set of "meta-rules". Meta-rules contain the generic information required to generate specific rule instances. The meta-rule associated with maneuver battalions in assembly area takes the following form:

```
(defargument (MB4-in-AA BATT#)
  ((MB4-AA-distance C1 C2)
   (MB4-AA-distance C2 C3)
   (MB4-AA-distance C3 C4)
   (MB4-AA-distance C4 C1)
   (right-angle C2 C1 C4)
   (right-angle C1 C4 C3))
(((MB4-in-AA BATT# C1 C2 C3 C4) y)))
```

This meta-rule essentially says that four companies satisfying the doctrinal requirements for maneuver battalions in assembly area form such a battalion with belief y. BATT# becomes the identifier for that battalion with its associated list of companies.

Now the list of premises is matched with the appropriate meta-rule to form a set of possible rules. For instance the meta rule identifiers C1 - C4 could be *bound* to companies A1 - A4 respectively. The generated rule takes the following form:

```
(defargument (MB4-in-AA BATT#)
  ((MB4-AA distance A1 A2)
   (MB4-AA-distance A2 A3)
```

```
    (MB4-AA-distance A3 A4)
    (MB4-AA-distance A4 A1)
    (right-angle A2 A1 A4)
    (right-angle A1 A4 A3))
 (((MB4-in-AA BATT# A1 A2 A3 A4) z1)))
```

For this simple case there is no more to be done. However, this would not normally be the only possibility. Consider a case where there are eight maneuver companies in close proximity (A1 through A8). The pre-processor groups "close" companies into a set ("close" is deemed to be within 2 kilometers). The premise list for the eight companies is formed as in the above example, but is much longer. Furthermore two distinct possibilities exist: (1) Two battalions consisting of four companies each (see Figure 4-9); (2) One battalion of five companies and one of three companies (see Figure 4-10).

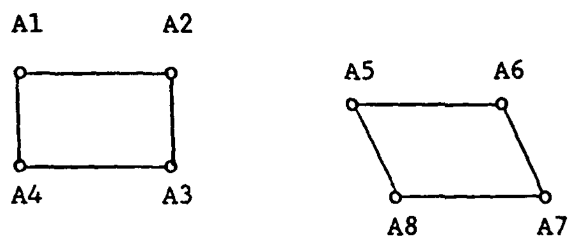


Figure 4-9. Eight maneuver companies in assembly area in a 4-4 combination.

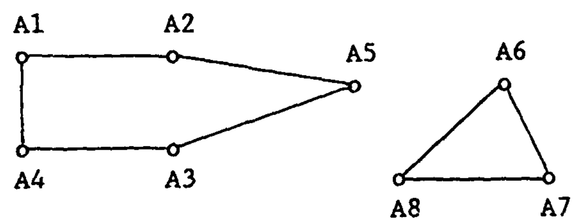


Figure 4-10. Eight maneuver companies in assembly area in a 5-3 combination.

The rule generator develops two rules accordingly. As the number of "close" companies increases, so does the number of possible premise and meta-rule sets, which in turn increases the number of rules.

At this stage, the rules are passed to the BMS along with a set of solution rules. For the four company case given above the solution rule takes the form:

```
(defargument BATT#1)
  ((MB4-in-AA BATT#1 A1 A2 A3 A4))
(((SOLUTION (BATT#1 MB4-in-AA)) m1)))
```

Notice that the antecedent for the solution rule is the same as the consequent for the rule generated from the premise and meta-rule. This is indicative of the tree structure underlying the rule-base.

The belief value m_1 is assessed by an expert for each solution rule. In the example with eight companies there are two solution rules (one for each possible combination of battalion formations). The solution rules take the form:

```
(defargument BATT#1 BATT#2
  ((MB4-in-AA BATT#1 A1 A2 A3 A4)
   (MB4-in-AA BATT#2 A5 A6 A7 A8))
(((SOLUTION (BATT#1 MB4-in-AA)(BATT#2 MB4-IN-AA)) m2)))
```

and:

```
(defargument BATT#3 BATT#4
  ((MB5-in-AA BATT#3 A1 A2 A3 A4 A5)
  ((MB3-in-AA BATT#4 A6 A7 A8))
(((SOLUTION (BATT#3 MB5-in-AA)(BATT#4 MB3-IN-AA)) m3)))
```

The BMS processes all the rules and solution rules and outputs solutions (as indicated on the "SOLUTION" line of the above solution rules). The solutions come complete with a degree of belief and a degree of conflict. For instance, in the first example companies A1-A4 form a battalion (labeled BATT#1) with a degree of belief and a degree of conflict calculated in the BMS. Details of the procedure for calculating beliefs can be found in Section 4.2.

The second example produces conflicting belief in the two possible combinations of battalion formations. This conflict is resolved in the conflict resolution process by "backing down" one of the possibilities - in this case, by backing down the combination of a 5-company battalion and a 3-company battalion in favor of the more likely combination of two 4-company battalions.

At this stage, we have only considered two very simple cases; one of which produces conflict. Normally scenarios can be expected to contain approximately 100 company-sized elements (most of which will be maneuver or artillery companies). The number of companies contained in a "close" set could increase substantially. This could have the effect of causing more cases, and higher degrees, of this type of conflict.

Furthermore, our examples have not considered extra-image features such as terrain, map-to-image, or near edge of the image. Recognition of these features causes more rule possibilities. Consider again our first example with four maneuver companies in an assembly area, and suppose two of the companies (A3 and A4) are near the edge of the section of the image being analyzed (see Figure 4-11). From this information the following solution rules can be generated:

```
(defargument BATT#1 NEAR-EDGE#1
  ((MB3-in-AA BATT#1 A1 A2 A3)
  ((C-in-AA NEAR-EDGE#1 A4))
(((SOLUTION (BATT#1 MB3-in-AA)(NEAR-EDGE#1 C-in-AA)) m4)))
```

and:

```
(defargument BATT#2 NEAR-EDGE#1
  ((MB3-in-AA BATT#2 A1 A2 A4)
  ((C-in-AA NEAR-EDGE#1 A3))
(((SOLUTION (BATT#2 MB3-in-AA)(NEAR-EDGE#1 C-in-AA)) m5)))
```

The second possibility is illustrated in Figure 4-11 by the dotted lines. As more possibilities are included, the number of solution rules increases, creating more conflict (even the original battalion of size 4 could be a battalion of size 5 if it were close enough to the edge of the image section).

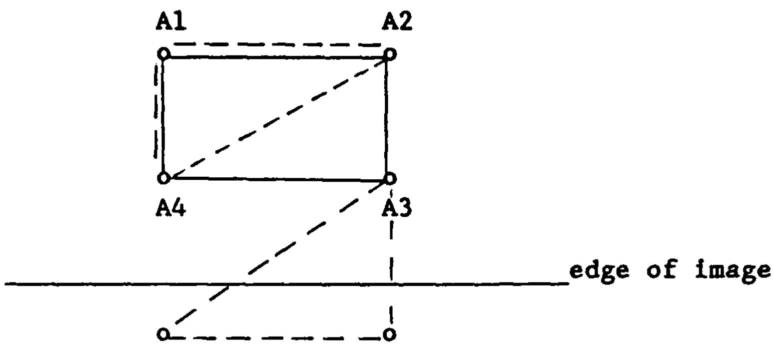


Figure 4-11. Four maneuver companies in assembly area with two companies"Near-Edge".

After receiving solutions from the BMS, CoMMiTR evaluates the degree of conflict. If the degree of conflict is unsatisfactory (greater than a threshold value) then the conflict resolution mechanism is activated. This looks at the causes of the conflicting information. For instance, in the example with eight companies given earlier, the conflict resolution mechanism may automatically back down the second solution rule on grounds of doctrinal formations. However, in cases such as this the conflict can also be pointed out to the user who may then indicate the direction of the conflict resolution. This process is explained more fully in Section 5.2. In other cases of conflict the "near-edge" characteristic may be adjusted or map-to-image registration may be questioned.

During the course of conflict resolution, CoMMiTR may pass out messages or suggestions about the causes of the conflict. For instance, it may suggest a review of the map-to-image module or a review of the descriptor set module. These messages are included in the final output (see Appendix C). In a production system they would cause calls to other systems.

When CoMMiTR has resolved as much conflict as it can, the final solution is presented in the form of an ASCII file that can be viewed in the available editor. The inputs and outputs used in CoMMiTR are described more fully in Section 5.

CoMMiTR's current rulebase consists of rules for integrating information from several low-level processing modules. Thus, CoMMiTR can be viewed as processing from a bottom-up perspective. However, as noted earlier, we believe that the most promising approach to image interpretation involves feedback cycles between systems operating at different levels of analysis. To implement this within the framework outlined in this report might involve running CoMMiTR separately on different slices of imagery from adjacent areas of the battlefield, analyzing the output (again within CoMMiTR) using higher-level rules about overall organization of the battlefield, and then cycling the results back to be reprocessed at the lower level. If this were done, unresolved conflict at the lower levels might be resolved by the higher level order-of-battle knowledge. This pyramidal approach is feasible combinatorically and, we believe, quite promising.

5.0 USING CoMMiTR

This section provides the procedures for executing CoMMiTR. CoMMiTR is designed to *commit* to a consensus among different automated image interpretation systems within the domain of Military Target Recognition. This section is intended to provide the necessary information to understand the operation of CoMMiTR and to employ CoMMiTR effectively. CoMMiTR functions are detailed in terms of their required inputs and outputs. Installation and machine requirements for CoMMiTR are also provided.

5.1 CoMMiTR INPUTS

There are essentially two types of input required by CoMMiTR. The first is initial input that includes the scenario, and the "worry" list. The second involves user input to the conflict resolution process.

CoMMiTR requires a scenario and a "worry" list for operation. A scenario is represented by two scenario files. The first contains information about each of the clusters of companies in a particular area. Appendix B contains sample scenarios and their pictoral representations on a stylized map. These scenarios were developed on a scenario generator developed on DSC's Symbolics computer, but scenarios can also be generated by entering image coordinates of features directly into a scenario file. Each company cluster is identified by its location, type, and posture. A typical company cluster takes the following form in a scenario file:

```
(make-company
          :ID     <company-ID>
          :x      <x-coordinate>
          :y      <y-coordinate>
          :z      <z-coordinate>
          :type   <company-type>
          :posture <company-posture>)
```

The items in brackets are arguments that must be supplied by the user. The ID is simply a unique identifier for a particular company. The coordinates are self-explanatory[2] and the company type and posture can take values as indicated in Tables 5-1 and 5-2.

TABLE 5-1. Company Types.

Maneuver (Tank or Motorized Rifle)
Artillery (Self Propelled)
SAM
FROG
Signal
Engineer
Unidentified

TABLE 5-2. Company Postures.

In Assembly Area
Field Emplaced
Moving to Contact
In Convoy

The second scenario file, the terrain data file, must include details of the impact of certain terrain (and other miscellaneous) features on the scenario. These include information about terrain features such as rivers, roads, relative elevation, and lakes, and about image features such as "near edge" and map to image registration. For example, a river is represented as a sequence of lines between points on the map. To define a river that starts at coordinates (26 1), continues to (26 3) and then to (25 4), the following is entered into the terrain data file:

```
(make-terrain-feature
  :id 'river
  :coordinates '((26.0 1.0) (26.0 3.0)))

(make-terrain-feature
  :id 'river
  :coordinates '((26.0 3.0) (25.0 4.0)))
```

----
2. This coordinate is a requirement of the LISP system used to develop CoMMiTR. It must always be set to zero when generating new company clusters.

Ridge lines can be specified by entering 'ridge-line in the :id field.

A full scenario, then, is specified by two files: a scenario file that defines the company-sized clusters on the image, and a terrain data file. To make a new scenario, the user must create a scenario file and a terrain data file, and then tell CoMMiTR that the two belong together as a scenario. This final step is performed by adding a new scenario to CoMMiTR's scenario list. To do this, the scenario creator must edit the CoMMiTR source file LOCAL-FI.LSP. The first Lisp expression in that file defines a constant called *scenario-alist*. A list containing the scenario's name and its two associated files must be added to *scenario-alist*. The scenario list that comes with the system is the following:

```
(defconstant *scenario-alist*
          '((scen1 "sc1.lsp" "terdata.lsp")
            (scen2 "sc2.lsp" "terdata.lsp")
            (scen3 "sc3.lsp" "terdata.lsp")
            (scen4 "sc4.lsp" "terdata.lsp")))
```

To add a new scenario with name "new-scen" with scenario files NEWSC.LSP and NEWTER.LSP, change this expression to

```
(defconstant *scenario-alist*
          '((scen1 "sc1.lsp" "terdata.lsp")
            (scen2 "sc2.lsp" "terdata.lsp")
            (scen3 "sc3.lsp" "terdata.lsp")
            (scen4 "sc4.lsp" "terdata.lsp")
            (new-scen "newsc.lsp" "newter.lsp")))
```

(Be sure all parentheses are matched when the new scenario is inserted).

Particular concerns must also be identified in the input file, such as all maneuver battalions in assembly area. These concerns are implemented on a "worry" list. Any number of worries may be specified concurrently for a particular scenario. Tables 5-1 and 5-2 indicate the possible specifications of company types and postures in the "worry" list. Particular (rectangular) areas of the image can be examined by inputting coordinates for the corresponding upper left and lower right corners. A typical worry list may take the form:

```
(((Maneuver) (convoy))
 ((Maneuver) (assembly-area)))
```

This worry list indicates a request for analysis of all maneuver battalions that are in convoy or in assembly area. If a narrower search was required, such as all maneuver battalions contained in the image that are within 10 kilometers of the FEBA, this would be implemented as:

```
(((Maneuver) 0 0 30 5))
```

This worry list specifies an analysis of all maneuver battalions in the region defined by the two coordinate pairs (0, 0) and (30, 5). As the FEBA is defined along a line approximately 5k north of the x-axis this effectively asks for all maneuver battalions in the image that are within 10 kilometers of the FEBA.

After logging in to the relevant account, CoMMiTR is invoked by typing "DEMO" at the terminal. Initially this invokes the available editor, which enables input to CoMMiTR via an input file. CoMMiTR takes as its input a file named INPUT.LSP. The editor is invoked on the most recent version of this file. The blank lines of the input file must be completed as indicated in the instructions. A sample input file is shown below. (Note: The user inputs are on the lines without semicolons in the first column.)

```
; On the next line that is not a comment (does not start with a semicolon),
; enter the name of the scenario that you wish to process.
; The avaliable scenarios are:
;    SCEN1
;    SCEN2
;    SCEN3
;    SCEN4
;
scen1
;
; On the following noncomment lines, enter the problems that you want the
; system to worry about. The problems should be a list (in parentheses) of a
; list of the types of companies that you are worried about, optionally
; followed by the coordinates that define a rectangle around the area you are
; worried about. For example, if you are worried about SAM's in a retangular
; area bound by (20 0) in the upper left corner and (23 7) in the lower right
; corner, your entry should be:
;      ((SAM) 20 0 23 7)
```

```
; If you are worried about artillery and engineers anywhere in the image area,
; your entry should be:
;      ((ART ENG))
;
; If you are worried about all types of companies, do not list any types.  The
; most general entry (worry about everything) is:
;
;      (())
;
; There may be as many entries as you wish.  The system will consider any
; problem that meets any of the worry conditions.
;
; The possible company types are:
;      ENG  "Engineer"
;      ART  "Artillery"
;      SAM  "SAM"
;      MAN  "Maneuver"
;      FROG "FROG"
((sam) 20 0 23 7)
;
```

After completing the required input simply type "^Z" followed by "exit" to save the input file. CoMMiTR now processes the selected scenario. In this example, CoMMiTR will process the scenario named SCEN1, and look for all SAM sites in the region defined by the coordinate pairs (20 0) (23 7).

In some instances further input may be required during processing to help CoMMiTR's conflict resolution process. When this happens two or more conflicting battalion formations are presented chronologically. A particular battalion formation is selected by choosing the appropriate number. CoMMiTR then continues processing. When CoMMiTR has completed conflict resolution the resulting battalion formations are output in a file "OUTPUT.TXT". The system then automatically invokes the editor for perusal of the results.

5.2 CoMMiTR OUTPUTS

The output file can be read by the available editor (it is in the form of an ASCII file). It contains the solution list of battalion formations generated by CoMMiTR. A part of the output file corresponding to Scenario 2 (see Appendix B) is provided below. The full output file is given in Appendix C.

```
The following battalion was selected with belief 0.5189047 and conflict 0.0
battalion made up of:
```

```
Company COMPANY-24 (a Maneuver company in Assembly area posture) at 11.55 4.45
Company COMPANY-23 (a Maneuver company in Assembly area posture) at 11.55 4.99
Company COMPANY-21 (a Maneuver company in Assembly area posture) at 10.95 4.86
Company COMPANY-22 (a Maneuver company in Assembly area posture) at 10.98 4.56
```

The solution list indicates the company clusters that comprise the higher level formations, and provides coordinates for, and type and posture of, those company clusters. The solution list also indicates the degrees of belief and conflict associated with each particular battalion formed as part of the solution.

The full solution list also contains messages that are passed out by CoMMiTR as suggestions for reanalysis of the low-level component systems. For example, the following output (again from Scenario 2) suggests a problem with map to image registration.

```
There may be map to image registration problems:
 DTED indicates company on impossible terrain for
   Company COMPANY-55 (a Artillery company in Field emplaced posture) at 19.14 1.76
```

Execution of CoMMiTR is completed, and the output file is saved, by typing "^Z" followed by "quit" at the terminal.

5.3 HARDWARE AND SOFTWARE REQUIREMENTS

CoMMiTR is contracted to run on a VAX machine with a COMMON LISP compiler. There are no special software features prohibiting installation on any computer with full COMMON LISP capabilities. There are no special requirements for the user interface, though CoMMiTR is implemented for a VT100 screen. CoMMiTR is not tied to any hardware or software environments other than those described here, hence it is extremely transportable.

6.0 DISCUSSION AND FURTHER WORK

This report described an implementation of the Non-Monotonic Probabilist inference engine and a system built around it, the CoMMiTR (Consensus Module for Military Target Recognition). Non-Monotonic Probabilist embeds a numerical uncertainty calculus in a qualitative reasoning process that allows making assumptions and revising them in response to conflict. The CoMMiTR system illustrates this capability in reasoning with the (simulated) outputs from different image processing modules to achieve a consensus interpretation of the image.

The present research could be extended in a number of directions. First is to develop an easier-to-use interface between NMP and the knowledge engineer, that would allow creating and modifying rule bases in a more conversational format, without the use of Lisp-like syntax. (How to develop a rule base in the present implementation is covered in Appendix A.) A second direction is further development of the inference capability itself. In particular, the present implementation sacrificed speed for flexibility and generality, and execution time can be slow for complex rulebases. Speeding up execution could be achieved by some combination of the following methods: partitioning the rule base; developing specialized structures to exploit special case models (characterized by independence between different lines of argument); and developing ways to combine our symbolic propagation with numerical propagation algorithms developed by others. A final research direction is further development of the CoMMiTR application. This would involve further knowledge engineering to validate and extend the rulebase, and developing the necessary links to implement information flows between CoMMiTR and other systems at ETL.

7.0 REFERENCES

Cohen, M.S., Watson, S.R., and Barrett, E. *Alternative theories of inference in expert systems for image analysis* (Technical Report 85-1). Falls Church, VA: Decision Science Consortium, Inc., January 1985.

deKleer, J. An assumption-based truth maintenance system. *Artificial Intelligence*, 1986a, *29*, 127-162.

deKleer, J. Extending the ATMS. *Artificial Intelligence*, 1986b, *28*, 163-196.

deKleer, J. Problem solving with the ATMS. *Artificial Intelligence*, 1986c, *28*, 197-224.

Laskey, K.B. *Integrating high and low level image processing*. DSC Working paper; Reston, VA: Decision Science Consortium, Inc., 1988.

Laskey, K.B., Cohen, M.S., and Martin, A.W. Representing and eliciting knowledge about uncertain evidence and its implications. *IEEE Transactions on Systems, Man, and Cybernetics*, 1988, to appear.

Laskey, K.B., and Lehner, P.E. Belief maintenance: An integrated approach to uncertainty management. *Proceedings of the Seventh National Conference on Artificial Intelligence*, August 1988.

Laskey, K.B., and Lehner, P.E. Assumptions, beliefs and probabilities. *Artificial Intelligence*, in press.

Pearl, J. Fusion, propagation and structuring in belief networks. *Artificial Intelligence*, 1986, *29*(3), 241-288.

Pearl, J. *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. San Mateo, CA: Morgan Kaufmann Publishers, Inc., 1988.

Rosenfeld, A., Hummel, R.A., and Zucker, S.W. Scene labeling by relaxation operations. *IEEE Transactions on Systems, Man and Cybernetics*, 1976, *SMC-6*, 420-433.

Rumelhart, D.E., McClelland, J.L., and the PDP Research Group, *Parallel Distributed Processing, Vol. I*, Cambridge, MA: MIT Press, 1986.

Shafer, G., and Tversky, A. Languages and designs for probability judgment. *Cognitive Science*, 1985, *9*, 309-339.

Toulmin, S., Rieke, R., and Janik, A. *An introduction to reasoning* (2nd Edition). NY: Macmillan Publishing Company, 1984.

APPENDIX A. USING NMP TO BUILD AN EXPERT SYSTEM

The non-monotonic probabilist itself is a generic tool for building expert systems. This appendix describes how to write rules in NMP, and illustrates it in the context of an example taken from Laskey and Lehner (in press).

A.1 DESCRIPTION OF THE RULE FORMAT

The rules for systems based on the Non-Monotonic Probabilist (NMP) are in the form of Lisp statements. The system does not require that knowledge engineers know how to program in Lisp, but allows those who do to use the full power of the language. A rulebase can be typed into a file using any standard text editor. A "Lisp-aware" editor, such as Zmacs on Symbolics systems, is helpful for matching the many parentheses in the rulebase, but is not required. The file may contain comments, which start with a semicolon and continue to the end of a line.

New-argument is usually the *first statement in a rulebase*. This statement reinitializes the NMP system. Its form in the file is:

```
(new-argument)
```

Like each statement in the rulebase, new-argument is enclosed in a set of parentheses.

Defhypset is the statement used to set up ATMS nodes as negations of one another. A typical rulebase will have defhypset statements to define each of the nodes used in the antecedents and consequences of its rules. These statements usually follow the new-argument statement and precede the rule definitions. The form of this statement is:

```
(defhypset (<node-name> <not-node-name>))
```

Defhypset requires a set of parentheses around the pair of node names in addition to the parentheses around the entire statement.

Defargument is the statement used to actually set up a rule. In addition to the antecedent, consequence, and belief that are shown in the rules in Depravia example, the system requires that each rule have a name. This name may be a descriptive phrase enclosed in parentheses, or simply a rule number. The name is used to refer to the rule during conflict resolution. The form of the defargument statement is:

```
(defargument <name> (<antecedent>) ((<consequent> <belief>)))
```

Defargument needs a set of parentheses around the antecedents, which may be more that one ATMS node, and two sets of parentheses around the consequent-belief pair. Again, a set of parentheses surround the entire statement. The defargument statement automatically creates belief tokens that the rule needs, assigns the proper probabilities to them, and creates the proper ATMS justification.

A.2 DEPRAVIA EXAMPLE

Relations between Depravia and Rechtia have been plagued by recurrent border disputes. Because these countries are of such strategic importance, concern has arisen over a recent report of increased activity in Depravia near the border area, which may be indicative of an impending attack. Figure A-1 illustrates some hypothetical inference rules for reasoning about Depravia's attack plans. Section A.3 shows how these inference rules are communicated to NMP.

A.3 NMP RULES FOR DEPRAVIA EXAMPLE

```
;;; The following is an actual example of how the Depravia example rulebase
;;; would be entered into the system.

(new-argument)     ; reinitialize the system

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; The following statements define the antecedent and consequent nodes for
;;; all of the rules. The NMP allow the user to use names of any length,
;;; so we are using the full names used in the English rules rather than the
;;; single letters used in the explanation.
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
(defhypset (Moving-troops-to-border Not-moving-troops-to-border))

; Corresponds to steps for Rule 1 antecedent

(defhypset (Attack-planned No-attack-planned))

; Corresponds to steps for Rule 1 consequent.

(defhypset (Readying-supply-lines Not-readying-supply-lines))

; Corresponds to steps for Rule 2 antecedent.

(defhypset (Increased-activity No-increased-activity))

; Corresponds to steps for Rule 3 antecedent.

(defhypset (Report-of-increased-activity No-report-of-increased-activity))

; Corresponds to steps for Rule 5 antecedent.

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; The following statements define the rules themselves.
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(defargument 1 (Moving-troops-to-border) ((Attack-planned .7)))

(defargument 2 (Readying-supply-lines) ((Attack-planned .8)))

(defargument 3 (Increased-activity) ((Moving-troops-to-border .6)))

(defargument 4 (Increased-activity) ((Readying-supply-lines .75)))

(defargument 5 (Report-of-increased-activity) ((Increased-activity .8)))
```

A.4 USING THE RULES TO DO INFERENCE IN NMP

After a rule set has been completed, Lisp statements are used to tell the system the facts that are known and to assess the belief in the propositions.

Load is the statement used to load a rule base file into the system. It is the same command used to load a Lisp program. The form of the load command is:

```
(load <filename>)
```

The load statement should be typed directly into the Lisp system. In most Lisp systems, there is no need to press the return key; when the parentheses are balanced, the command is executed. If the rules for the depravia example were in a file called "DEPRAVIA", the user would type:

```
(load "DEPRAVIA")
```

at the Lisp prompt to load the rule set.

Premise is the statement that enters evidence into the system. Each premise command typed into the system enters a single piece of evidence. It is typed into the system as:

```
(premise '<evidence>)
```

It is important to note that the premise command requires a single quote character before the evidence. To enter the evidence for the Depravia example, the user would type:

```
(premise 'Report-of-increased-activity)
```

This would set the label of the ATMS node Report-of-increased-activity to the empty environment, which indicates that Report-of-increased-activity is always true.

Datum-belief is the statement that is used to assess belief in a proposition. The datum-belief statement will return the belief that the evidence implies the proposition and a measure of the conflict associated with the calculation. These values are printed to the screen if the statement is used alone, or available for use in further calculations if used in a more complex Lisp expression. The form of the datum-belief statement is:

```
(datum-belief <proposition>)
```

To calculate the belief in the proposition Attack-planned in the Depravia example, the user would type:

```
(datum-belief Attack-planned)
```

The system would print to the screen the results of the belief calculation, 0.614, and a conflict factor of 0.0, indicating that there was no conflict in the calculation.

Inference Rules and Corresponding ATMS Justifications

Inference Rule	**ATMS Justification**
Moving_Troops_To_Border $\xrightarrow[(.7)]{}$ Attack_Planned	$b, V \Rightarrow a$
Readying_Supply_Lines $\xrightarrow[(.8)]{}$ Attack_Planned	$c, W \Rightarrow a$
Increased_Activity $\xrightarrow[(.6)]{}$ Moving_Troops_to_Border	$d, X \Rightarrow b$
Increased_Activity $\xrightarrow[(.75)]{}$ Readying_Supply_Lines	$d, Y \Rightarrow c$
(Report Increased_Activity) $\xrightarrow[(.8)]{}$ Increased_Activity	$e, Z \Rightarrow d$

Auxiliary Hypotheses Definitions	**Additional ATMS Justifications**
$pdist\{V,\neg V;\ .7\ .3\}$	$a, \neg a \Rightarrow \bot$
$pdist\{W,\neg W;\ .8, .2\}$	$b, \neg b \Rightarrow \bot$
$pdist\{X,\neg X;\ .6, .4\}$	$c, \neg c \Rightarrow \bot$
$pdist\{Y,\neg Y;\ .75, .25\}$	$d, \neg d \Rightarrow \bot$
$pdist\{Z,\neg Z;\ .8, .2\}$	$e, \neg e \Rightarrow \bot$

Diagram of the Inference Network

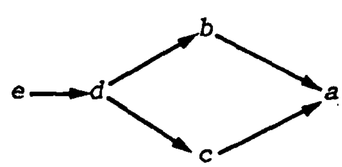


Figure A-1. Intelligence analysis example.

APPENDIX B. CoMMiTR SCENARIOS

This appendix describes the scenarios implemented in the initial version of CoMMiTR. The four scenarios were designed to cover the major features of the CoMMiTR program while representing the salient features of a successful attacking force. These scenarios are viewed as simulated SAR images taken sequentially with a time interval of approximately 3 hours. The scenario pictures cover an area approximately 30k long by 10k deep, and approximately 5k back from the FEBA (i.e., behind enemy lines).

The four scenarios are depicted in Figures B-1 through B-4 below (the x-axis in the figures corresponds to the line approximately 5k behind the FEBA). These figures were created with a scenario generator built specifically for application with CoMMiTR[3]. The scenarios are also shown in coded form in Figures B-5 through B-8. These are simply LISP files that are accessed by CoMMiTR when they are selected for demonstration. Any new scenarios can be created in this form to be used with CoMMiTR. The remainder of this appendix provides brief descriptions of the four scenarios.

Scenario-1: This scenario represents forces behind the front lines in a relatively static sector. Approximately three Regiments of maneuver elements, fire artillery battalions, and an engineer battalion are in the 1st echelon deep belt.

Scenario-2: This scenario shows that an artillery concentration has occurred for deep penetration and breakout, and that forces are concentrating for a push. Note that engineer assets are concentrating along the axis of advance. There is a major road through their location. Note also the SAM protection of the 2nd echelon forces.

----

3. The scenario generator was built to run on a Symbolics computer. However, the scenario generator software is compatible with the CoMMiTR software and could be adapted for another machine.

Scenario-3: This scenario shows the rarefaction as the enemy forces move forward. The maneuver elements are moving to contact and have moved out of range of the simulated SAR image.

Scenario-4: This scenario shows forces moving in to shore up the left flank of the penetration (which occurred off the map between image coordinates 14 to 26 on the horizontal grid).

Although the artillery battalions were not shown to move through the three time periods they could have jostled around to indicate friendly counter battery fire. However, this aspect of fire support was seen as tangential to the inferential process.

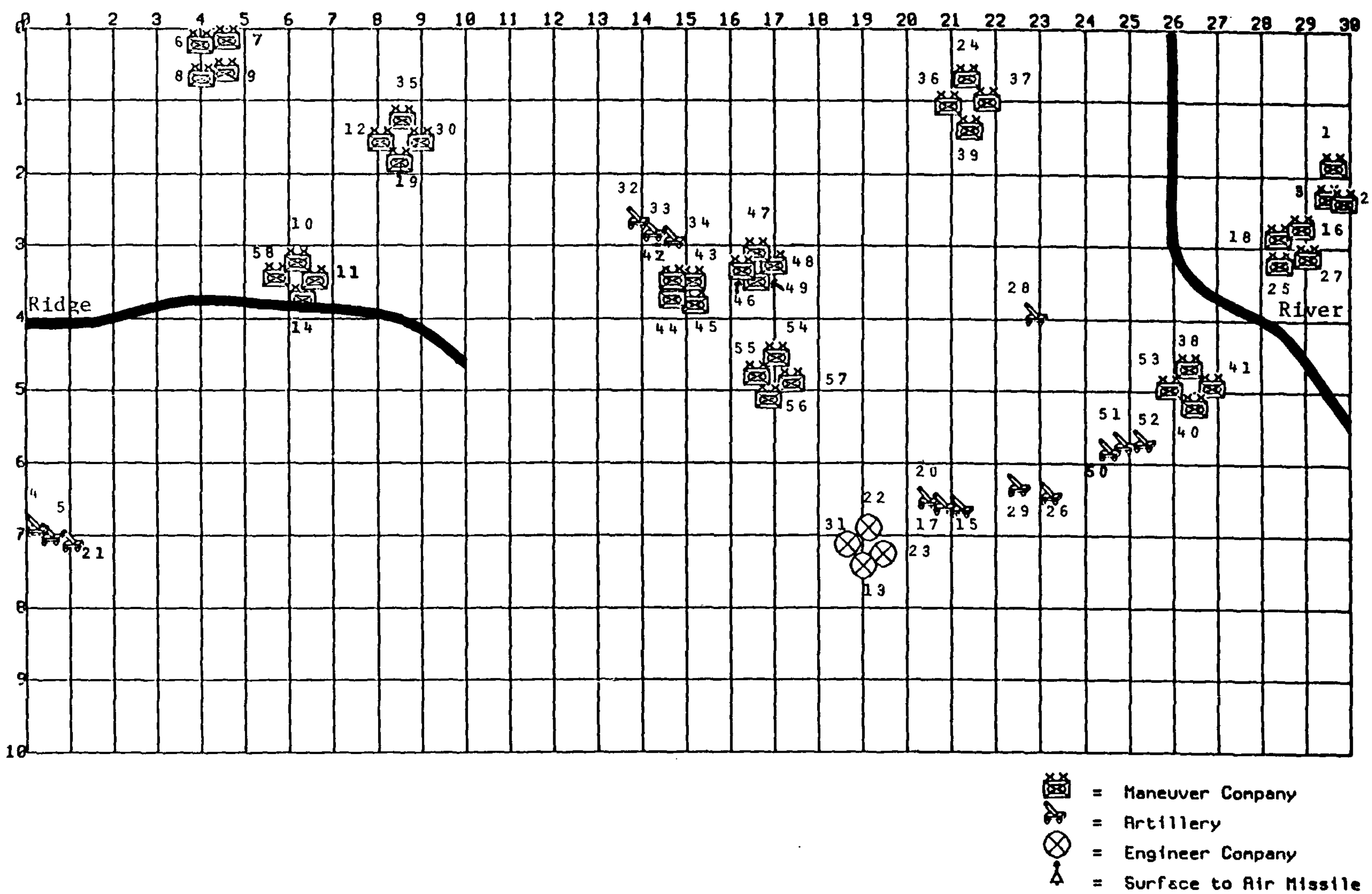


Figure B-1. Scenario 1.

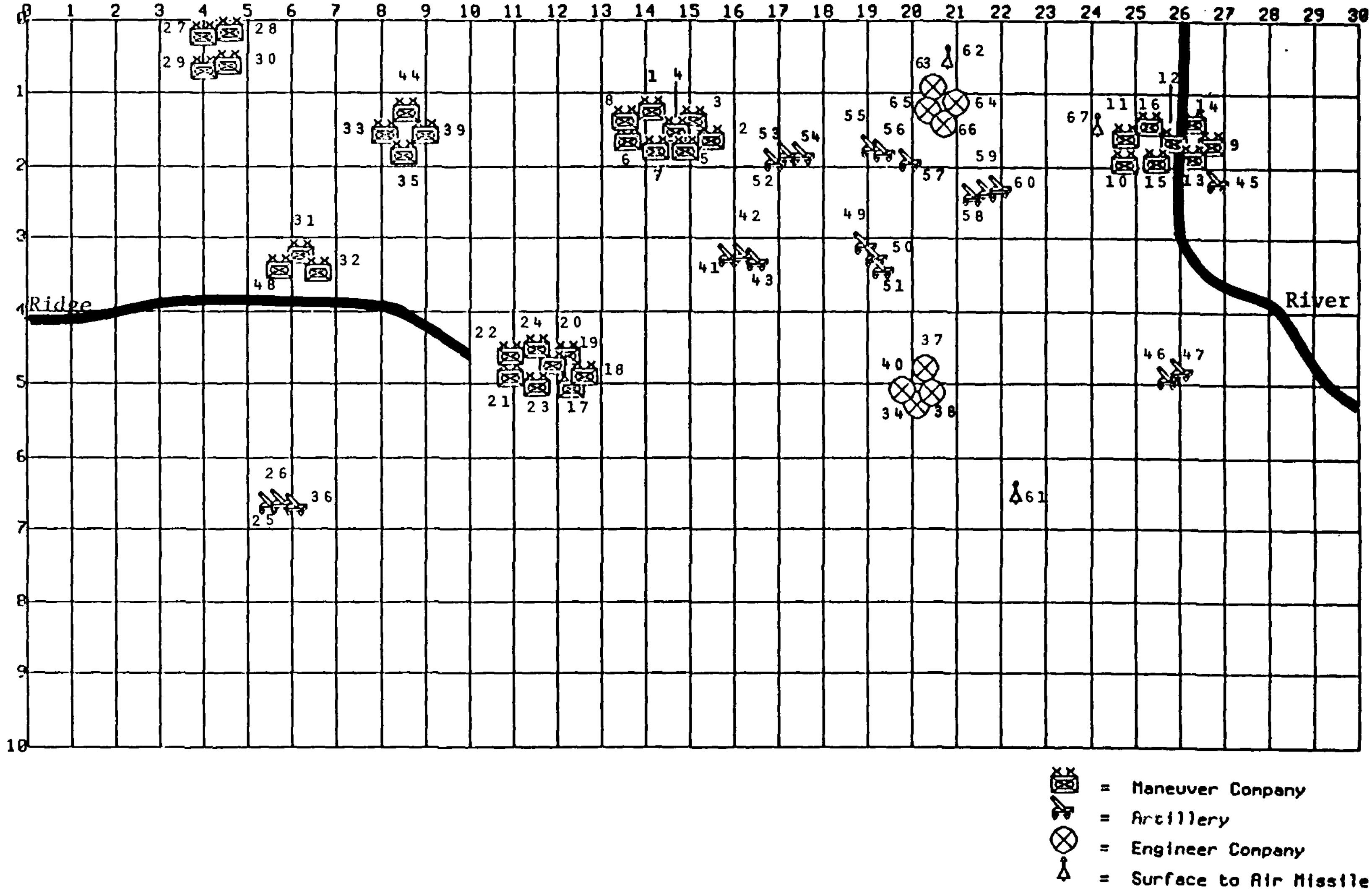


Figure B-2. Scenario 2.

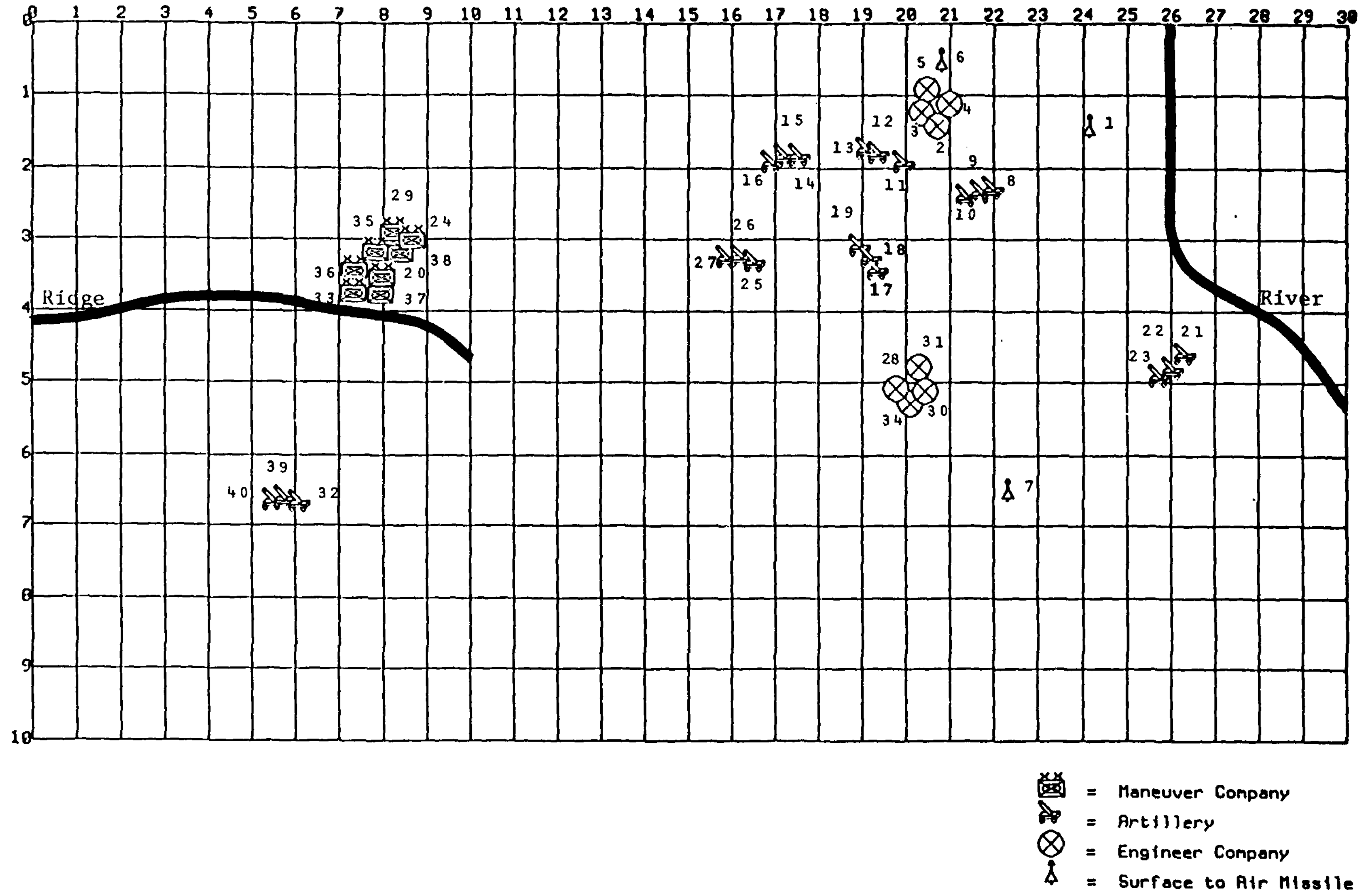


Figure B-3. Scenario 3.

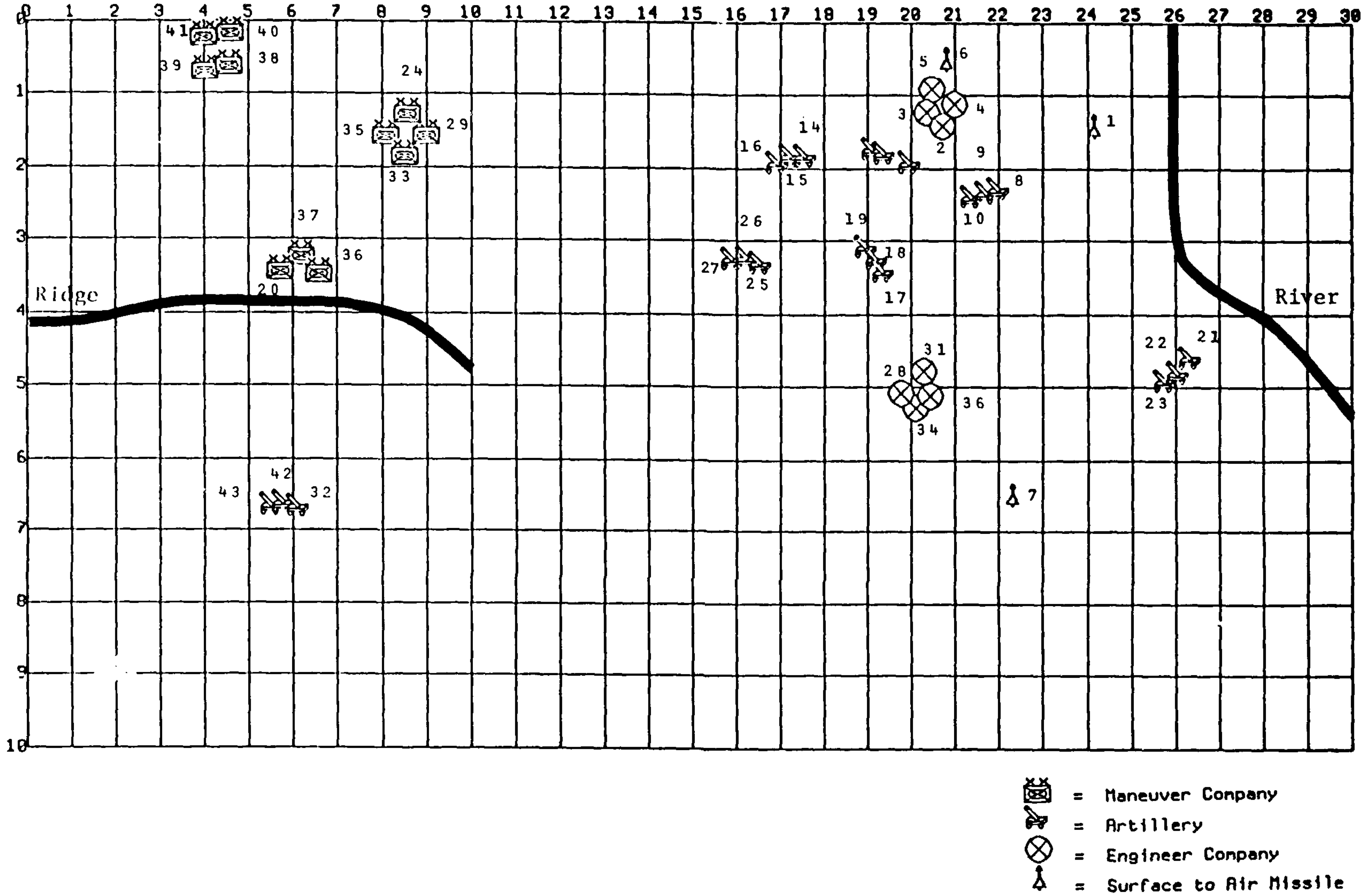


Figure B-4. Scenario 4.

```
(MAKE-COMPANY :X 29.64 :Y 1.81 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-1)
(MAKE-COMPANY :X 29.88 :Y 2.30 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-2)
(MAKE-COMPANY :X 29.52 :Y 2.24 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-3)
(MAKE-COMPANY :X 0.30 :Y 6.90 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-4)
(MAKE-COMPANY :X 0.64 :Y 7.02 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-5)
(MAKE-COMPANY :X 3.99 :Y 0.17 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-6)
(MAKE-COMPANY :X 4.60 :Y 0.11 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-7)
(MAKE-COMPANY :X 4.02 :Y 0.64 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-8)
(MAKE-COMPANY :X 4.57 :Y 0.56 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-9)
(MAKE-COMPANY :X 6.20 :Y 3.18 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-10)
(MAKE-COMPANY :X 6.59 :Y 3.42 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-11)
(MAKE-COMPANY :X 8.11 :Y 1.52 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-12)
(MAKE-COMPANY :X 19.02 :Y 7.43 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-13)
(MAKE-COMPANY :X 6.32 :Y 3.69 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-14)
(MAKE-COMPANY :X 21.29 :Y 6.64 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-15)
(MAKE-COMPANY :X 28.91 :Y 2.68 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-16)
(MAKE-COMPANY :X 20.90 :Y 6.60 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-17)
(MAKE-COMPANY :X 28.40 :Y 2.81 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-18)
(MAKE-COMPANY :X 8.53 :Y 1.80 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-19)
(MAKE-COMPANY :X 20.54 :Y 6.53 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-20)
(MAKE-COMPANY :X 1.12 :Y 7.11 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-21)
(MAKE-COMPANY :X 19.14 :Y 6.90 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-22)
(MAKE-COMPANY :X 19.48 :Y 7.26 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-23)
(MAKE-COMPANY :X 21.35 :Y 0.64 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-24)
(MAKE-COMPANY :X 28.43 :Y 3.18 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-25)
(MAKE-COMPANY :X 23.29 :Y 6.45 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-26)
(MAKE-COMPANY :X 29.06 :Y 3.09 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-27)
(MAKE-COMPANY :X 22.96 :Y 3.97 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-28)
(MAKE-COMPANY :X 22.56 :Y 6.34 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-29)
(MAKE-COMPANY :X 9.01 :Y 1.52 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-30)
(MAKE-COMPANY :X 18.66 :Y 7.13 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-31)
(MAKE-COMPANY :X 13.97 :Y 2.66 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-32)
(MAKE-COMPANY :X 14.34 :Y 2.82 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-33)
(MAKE-COMPANY :X 14.79 :Y 2.92 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-34)
(MAKE-COMPANY :X 8.59 :Y 1.22 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-35)
(MAKE-COMPANY :X 20.93 :Y 1.01 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-36)
(MAKE-COMPANY :X 21.81 :Y 0.95 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-37)
(MAKE-COMPANY :X 26.37 :Y 4.62 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-38)
(MAKE-COMPANY :X 21.41 :Y 1.35 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-39)
(MAKE-COMPANY :X 26.49 :Y 5.16 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-40)
(MAKE-COMPANY :X 26.89 :Y 4.88 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-41)
(MAKE-COMPANY :X 14.70 :Y 3.42 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-42)
(MAKE-COMPANY :X 15.15 :Y 3.44 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-43)
(MAKE-COMPANY :X 14.70 :Y 3.69 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-44)
(MAKE-COMPANY :X 15.21 :Y 3.76 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-45)
(MAKE-COMPANY :X 16.27 :Y 3.29 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-46)
(MAKE-COMPANY :X 16.60 :Y 3.05 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-47)
(MAKE-COMPANY :X 17.00 :Y 3.22 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-48)
(MAKE-COMPANY :X 16.60 :Y 3.44 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-49)
(MAKE-COMPANY :X 24.62 :Y 5.84 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-50)
(MAKE-COMPANY :X 24.95 :Y 5.74 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-51)
```

Figure B-5. Scenario file for first scenario.
(Page 1 of 2)

```
(MAKE-COMPANY :X 25.40 :Y 5.71 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-52)
(MAKE-COMPANY :X 25.92 :Y 4.92 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-53)
(MAKE-COMPANY :X 17.06 :Y 4.49 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-54)
(MAKE-COMPANY :X 16.60 :Y 4.75 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-55)
(MAKE-COMPANY :X 16.88 :Y 5.07 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-56)
(MAKE-COMPANY :X 17.39 :Y 4.85 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-57)
(MAKE-COMPANY :X 5.72 :Y 3.39 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-58)
```

Figure B-5. Scenario file for first scenario.
(Page 2 of 2)

```
(MAKE-COMPANY :X 14.15 :Y 1.20 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-1)
(MAKE-COMPANY :X 15.48 :Y 1.57 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-2)
(MAKE-COMPANY :X 15.12 :Y 1.33 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-3)
(MAKE-COMPANY :X 14.70 :Y 1.46 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-4)
(MAKE-COMPANY :X 14.94 :Y 1.78 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-5)
(MAKE-COMPANY :X 13.61 :Y 1.59 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-6)
(MAKE-COMPANY :X 14.24 :Y 1.74 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-7)
(MAKE-COMPANY :X 13.55 :Y 1.31 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-8)
(MAKE-COMPANY :X 26.77 :Y 1.63 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-9)
(MAKE-COMPANY :X 24.80 :Y 1.89 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-10)
(MAKE-COMPANY :X 24.80 :Y 1.55 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-11)
(MAKE-COMPANY :X 25.89 :Y 1.61 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-12)
(MAKE-COMPANY :X 26.34 :Y 1.83 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-13)
(MAKE-COMPANY :X 26.37 :Y 1.33 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-14)
(MAKE-COMPANY :X 25.50 :Y 1.87 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-15)
(MAKE-COMPANY :X 25.34 :Y 1.38 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-16)
(MAKE-COMPANY :X 12.34 :Y 5.01 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-17)
(MAKE-COMPANY :X 12.64 :Y 4.83 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-18)
(MAKE-COMPANY :X 12.28 :Y 4.55 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-19)
(MAKE-COMPANY :X 11.92 :Y 4.70 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-20)
(MAKE-COMPANY :X 10.95 :Y 4.86 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-21)
(MAKE-COMPANY :X 10.98 :Y 4.56 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-22)
(MAKE-COMPANY :X 11.55 :Y 4.99 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-23)
(MAKE-COMPANY :X 11.55 :Y 4.45 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-24)
(MAKE-COMPANY :X 5.53 :Y 6.68 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-25)
(MAKE-COMPANY :X 5.81 :Y 6.64 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-26)
(MAKE-COMPANY :X 3.99 :Y 0.17 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-27)
(MAKE-COMPANY :X 4.60 :Y 0.11 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-28)
(MAKE-COMPANY :X 4.02 :Y 0.64 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-29)
(MAKE-COMPANY :X 4.57 :Y 0.56 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-30)
(MAKE-COMPANY :X 6.20 :Y 3.18 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-31)
(MAKE-COMPANY :X 6.59 :Y 3.42 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-32)
(MAKE-COMPANY :X 8.11 :Y 1.52 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-33)
(MAKE-COMPANY :X 20.11 :Y 5.29 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-34)
(MAKE-COMPANY :X 8.53 :Y 1.80 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-35)
(MAKE-COMPANY :X 6.14 :Y 6.70 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-36)
(MAKE-COMPANY :X 20.29 :Y 4.79 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-37)
(MAKE-COMPANY :X 20.44 :Y 5.13 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-38)
(MAKE-COMPANY :X 9.01 :Y 1.52 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-39)
(MAKE-COMPANY :X 19.78 :Y 5.09 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-40)
(MAKE-COMPANY :X 15.94 :Y 3.27 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-41)
(MAKE-COMPANY :X 16.27 :Y 3.26 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-42)
(MAKE-COMPANY :X 16.57 :Y 3.33 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-43)
(MAKE-COMPANY :X 8.59 :Y 1.22 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-44)
(MAKE-COMPANY :X 26.92 :Y 2.21 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-45)
(MAKE-COMPANY :X 25.80 :Y 4.94 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-46)
(MAKE-COMPANY :X 26.10 :Y 4.83 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-47)
(MAKE-COMPANY :X 5.72 :Y 3.39 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-48)
(MAKE-COMPANY :X 18.99 :Y 3.11 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-49)
(MAKE-COMPANY :X 19.24 :Y 3.27 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-50)
(MAKE-COMPANY :X 19.39 :Y 3.44 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-51)
```

Figure B-6. Scenario file for second scenario.
(Page 1 of 2)

```
(MAKE-COMPANY :X 16.97 :Y 1.95 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-52)
(MAKE-COMPANY :X 17.27 :Y 1.85 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-53)
(MAKE-COMPANY :X 17.60 :Y 1.85 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-54)
(MAKE-COMPANY :X 19.14 :Y 1.76 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-55)
(MAKE-COMPANY :X 19.42 :Y 1.81 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-56)
(MAKE-COMPANY :X 19.99 :Y 1.95 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-57)
(MAKE-COMPANY :X 21.41 :Y 2.41 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-58)
(MAKE-COMPANY :X 21.75 :Y 2.36 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-59)
(MAKE-COMPANY :X 22.02 :Y 2.30 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-60)
(MAKE-COMPANY :X 22.32 :Y 6.51 :Z 0 :TYPE 'SAM :POSTURE 'FE :ID 'COMPANY-61)
(MAKE-COMPANY :X 20.81 :Y 0.52 :Z 0 :TYPE 'SAM :POSTURE 'FE :ID 'COMPANY-62)
(MAKE-COMPANY :X 20.48 :Y 0.92 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-63)
(MAKE-COMPANY :X 20.99 :Y 1.12 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-64)
(MAKE-COMPANY :X 20.35 :Y 1.23 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-65)
(MAKE-COMPANY :X 20.72 :Y 1.42 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-66)
(MAKE-COMPANY :X 24.16 :Y 1.44 :Z 0 :TYPE 'SAM :POSTURE 'FE :ID 'COMPANY-67)
```

Figure B-6. Scenario file for second scenario.
(Page 2 of 2)

```
(MAKE-COMPANY :X 24.16 :Y 1.44 :Z 0 :TYPE 'SAM :POSTURE 'FE :ID 'COMPANY-1)
(MAKE-COMPANY :X 20.72 :Y 1.42 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-2)
(MAKE-COMPANY :X 20.35 :Y 1.23 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-3)
(MAKE-COMPANY :X 20.99 :Y 1.12 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-4)
(MAKE-COMPANY :X 20.48 :Y 0.92 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-5)
(MAKE-COMPANY :X 20.81 :Y 0.52 :Z 0 :TYPE 'SAM :POSTURE 'FE :ID 'COMPANY-6)
(MAKE-COMPANY :X 22.32 :Y 6.51 :Z 0 :TYPE 'SAM :POSTURE 'FE :ID 'COMPANY-7)
(MAKE-COMPANY :X 22.02 :Y 2.30 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-8)
(MAKE-COMPANY :X 21.75 :Y 2.36 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-9)
(MAKE-COMPANY :X 21.41 :Y 2.41 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-10)
(MAKE-COMPANY :X 19.99 :Y 1.95 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-11)
(MAKE-COMPANY :X 19.42 :Y 1.81 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-12)
(MAKE-COMPANY :X 19.14 :Y 1.76 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-13)
(MAKE-COMPANY :X 17.60 :Y 1.85 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-14)
(MAKE-COMPANY :X 17.27 :Y 1.85 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-15)
(MAKE-COMPANY :X 16.97 :Y 1.95 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-16)
(MAKE-COMPANY :X 19.39 :Y 3.44 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-17)
(MAKE-COMPANY :X 19.24 :Y 3.27 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-18)
(MAKE-COMPANY :X 18.99 :Y 3.11 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-19)
(MAKE-COMPANY :X 7.98 :Y 3.48 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-20)
(MAKE-COMPANY :X 26.10 :Y 4.83 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-21)
(MAKE-COMPANY :X 25.80 :Y 4.94 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-22)
(MAKE-COMPANY :X 26.37 :Y 4.62 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-23)
(MAKE-COMPANY :X 8.68 :Y 2.96 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-24)
(MAKE-COMPANY :X 16.57 :Y 3.33 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-25)
(MAKE-COMPANY :X 16.27 :Y 3.26 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-26)
(MAKE-COMPANY :X 15.94 :Y 3.27 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-27)
(MAKE-COMPANY :X 19.78 :Y 5.09 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-28)
(MAKE-COMPANY :X 8.26 :Y 2.84 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-29)
(MAKE-COMPANY :X 20.44 :Y 5.13 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-30)
(MAKE-COMPANY :X 20.29 :Y 4.79 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-31)
(MAKE-COMPANY :X 6.14 :Y 6.70 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-32)
(MAKE-COMPANY :X 7.32 :Y 3.70 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-33)
(MAKE-COMPANY :X 20.11 :Y 5.29 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-34)
(MAKE-COMPANY :X 7.83 :Y 3.12 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-35)
(MAKE-COMPANY :X 7.35 :Y 3.39 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-36)
(MAKE-COMPANY :X 7.95 :Y 3.72 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-37)
(MAKE-COMPANY :X 8.41 :Y 3.14 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-38)
(MAKE-COMPANY :X 5.81 :Y 6.64 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-39)
(MAKE-COMPANY :X 5.53 :Y 6.68 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-40)
```

Figure B-7. Scenario file for third scenario.

```
(MAKE-COMPANY :X 24.16 :Y 1.44 :Z 0 :TYPE 'SAM :POSTURE 'FE :ID 'COMPANY-1)
(MAKE-COMPANY :X 20.72 :Y 1.42 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-2)
(MAKE-COMPANY :X 20.35 :Y 1.23 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-3)
(MAKE-COMPANY :X 20.99 :Y 1.12 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-4)
(MAKE-COMPANY :X 20.48 :Y 0.92 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-5)
(MAKE-COMPANY :X 20.81 :Y 0.52 :Z 0 :TYPE 'SAM :POSTURE 'FE :ID 'COMPANY-6)
(MAKE-COMPANY :X 22.32 :Y 6.51 :Z 0 :TYPE 'SAM :POSTURE 'FE :ID 'COMPANY-7)
(MAKE-COMPANY :X 22.02 :Y 2.30 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-8)
(MAKE-COMPANY :X 21.75 :Y 2.36 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-9)
(MAKE-COMPANY :X 21.41 :Y 2.41 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-10)
(MAKE-COMPANY :X 19.99 :Y 1.95 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-11)
(MAKE-COMPANY :X 19.42 :Y 1.81 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-12)
(MAKE-COMPANY :X 19.14 :Y 1.76 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-13)
(MAKE-COMPANY :X 17.60 :Y 1.85 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-14)
(MAKE-COMPANY :X 17.27 :Y 1.85 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-15)
(MAKE-COMPANY :X 16.97 :Y 1.95 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-16)
(MAKE-COMPANY :X 19.39 :Y 3.44 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-17)
(MAKE-COMPANY :X 19.24 :Y 3.27 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-18)
(MAKE-COMPANY :X 18.99 :Y 3.11 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-19)
(MAKE-COMPANY :X 5.72 :Y 3.39 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-20)
(MAKE-COMPANY :X 26.10 :Y 4.83 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-21)
(MAKE-COMPANY :X 25.80 :Y 4.94 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-22)
(MAKE-COMPANY :X 26.37 :Y 4.62 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-23)
(MAKE-COMPANY :X 8.59 :Y 1.22 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-24)
(MAKE-COMPANY :X 16.57 :Y 3.33 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-25)
(MAKE-COMPANY :X 16.27 :Y 3.26 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-26)
(MAKE-COMPANY :X 15.94 :Y 3.27 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-27)
(MAKE-COMPANY :X 19.78 :Y 5.09 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-28)
(MAKE-COMPANY :X 9.01 :Y 1.52 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-29)
(MAKE-COMPANY :X 20.44 :Y 5.13 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-30)
(MAKE-COMPANY :X 20.29 :Y 4.79 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-31)
(MAKE-COMPANY :X 6.14 :Y 6.70 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-32)
(MAKE-COMPANY :X 8.53 :Y 1.80 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-33)
(MAKE-COMPANY :X 20.11 :Y 5.29 :Z 0 :TYPE 'ENG :POSTURE 'ASM :ID 'COMPANY-34)
(MAKE-COMPANY :X 8.11 :Y 1.52 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-35)
(MAKE-COMPANY :X 6.59 :Y 3.42 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-36)
(MAKE-COMPANY :X 6.20 :Y 3.18 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-37)
(MAKE-COMPANY :X 4.57 :Y 0.56 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-38)
(MAKE-COMPANY :X 4.02 :Y 0.64 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-39)
(MAKE-COMPANY :X 4.60 :Y 0.11 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-40)
(MAKE-COMPANY :X 3.99 :Y 0.17 :Z 0 :TYPE 'MAN :POSTURE 'ASM :ID 'COMPANY-41)
(MAKE-COMPANY :X 5.81 :Y 6.64 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-42)
(MAKE-COMPANY :X 5.53 :Y 6.68 :Z 0 :TYPE 'ART :POSTURE 'FE :ID 'COMPANY-43)
```

Figure B-8. Scenario file for fourth scenario.

APPENDIX C. OUTPUT FILE FOR SCENARIO-2

```
The following battalions were selected with belief 0.011800503 and conflict 0.0
Battalion made up of:
    Company COMPANY-8 (a Maneuver company in Assembly area posture) at 13.55 1.31
    Company COMPANY-6 (a Maneuver company in Assembly area posture) at 13.61 1.59
    Company COMPANY-7 (a Maneuver company in Assembly area posture) at 14.24 1.74
    Company COMPANY-1 (a Maneuver company in Assembly area posture) at 14.15 1.2

Battalion made up of:
    Company COMPANY-2 (a Maneuver company in Assembly area posture) at 15.48 1.57
    Company COMPANY-5 (a Maneuver company in Assembly area posture) at 14.94 1.78
    Company COMPANY-4 (a Maneuver company in Assembly area posture) at 14.7 1.46
    Company COMPANY-3 (a Maneuver company in Assembly area posture) at 15.12 1.33

------------------------------------------------------------
The following battalions were selected with belief 0.00030502726 and conflict 0.0
Battalion made up of:
    Company COMPANY-10 (a Maneuver company in Assembly area posture) at 24.8 1.89
    Company COMPANY-15 (a Maneuver company in Assembly area posture) at 25.5 1.87
    Company COMPANY-12 (a Maneuver company in Assembly area posture) at 25.89 1.61
    Company COMPANY-16 (a Maneuver company in Assembly area posture) at 25.34 1.38
    Company COMPANY-11 (a Maneuver company in Assembly area posture) at 24.8 1.55

Battalion made up of:
    Company COMPANY-9 (a Maneuver company in Assembly area posture) at 26.77 1.63
    Company COMPANY-14 (a Maneuver company in Assembly area posture) at 26.37 1.33
    Company COMPANY-13 (a Maneuver company in Assembly area posture) at 26.34 1.83

------------------------------------------------------------
The following battalions were selected with belief 0.5189047 and conflict 0.0
Battalion made up of:
    Company COMPANY-24 (a Maneuver company in Assembly area posture) at 11.55 4.45
    Company COMPANY-23 (a Maneuver company in Assembly area posture) at 11.55 4.99
    Company COMPANY-21 (a Maneuver company in Assembly area posture) at 10.95 4.86
    Company COMPANY-22 (a Maneuver company in Assembly area posture) at 10.98 4.56

Battalion made up of:
    Company COMPANY-17 (a Maneuver company in Assembly area posture) at 12.34 5.01
    Company COMPANY-20 (a Maneuver company in Assembly area posture) at 11.92 4.7
    Company COMPANY-19 (a Maneuver company in Assembly area posture) at 12.28 4.55
    Company COMPANY-18 (a Maneuver company in Assembly area posture) at 12.64 4.83

------------------------------------------------------------
The following battalions were selected with belief 0.8445626 and conflict 0.0
Battalion made up of:
    Company COMPANY-30 (a Maneuver company in Assembly area posture) at 4.57 0.56
    Company COMPANY-29 (a Maneuver company in Assembly area posture) at 4.02 0.64
    Company COMPANY-27 (a Maneuver company in Assembly area posture) at 3.99 0.17
    Company COMPANY-28 (a Maneuver company in Assembly area posture) at 4.6 0.11

------------------------------------------------------------
```

The following battalions were selected with belief 0.71999997 and conflict 0.0
Battalion made up of:
    Company COMPANY-26 (a Artillery company in Field emplaced posture) at 5.81 6.64
    Company COMPANY-36 (a Artillery company in Field emplaced posture) at 6.14 6.7
    Company COMPANY-25 (a Artillery company in Field emplaced posture) at 5.53 6.68

The following battalions were selected with belief 0.22277442 and conflict 0.0
Battalion made up of:
    Company COMPANY-40 (a Engineer company in Assembly area posture) at 19.78 5.09
    Company COMPANY-37 (a Engineer company in Assembly area posture) at 20.29 4.79
    Company COMPANY-38 (a Engineer company in Assembly area posture) at 20.44 5.13
    Company COMPANY-34 (a Engineer company in Assembly area posture) at 20.11 5.29

The following battalions were selected with belief 0.5726295 and conflict 0.0
Battalion made up of:
    Company COMPANY-44 (a Maneuver company in Assembly area posture) at 8.59 1.22
    Company COMPANY-39 (a Maneuver company in Assembly area posture) at 9.01 1.52
    Company COMPANY-35 (a Maneuver company in Assembly area posture) at 8.53 1.8
    Company COMPANY-33 (a Maneuver company in Assembly area posture) at 8.11 1.52

The following battalions were selected with belief 0.23684208 and conflict 0.80999994
The following companies that do not form a battalion are:
    Company COMPANY-46 (a Artillery company in Field emplaced posture) at 25.8 4.94
    Company COMPANY-47 (a Artillery company in Field emplaced posture) at 26.1 4.83

The following battalions were selected with belief 0.45 and conflict 0.0
Possible battalion made up of:
    Company COMPANY-32 (a Maneuver company in Assembly area posture) at 6.59 3.42
    Company COMPANY-31 (a Maneuver company in Assembly area posture) at 6.2 3.18
    Company COMPANY-48 (a Maneuver company in Assembly area posture) at 5.72 3.39
and other 1 or more companies that do not appear on the image

----

The following battalions were selected with belief 0.0244433 and conflict 0.0
Battalion made up of:
    Company COMPANY-50 (a Artillery company in Field emplaced posture) at 19.24 3.27
    Company COMPANY-51 (a Artillery company in Field emplaced posture) at 19.39 3.44
    Company COMPANY-49 (a Artillery company in Field emplaced posture) at 18.99 3.11

Battalion made up of:
    Company COMPANY-53 (a Artillery company in Field emplaced posture) at 17.27 1.85
    Company COMPANY-54 (a Artillery company in Field emplaced posture) at 17.6 1.85
    Company COMPANY-52 (a Artillery company in Field emplaced posture) at 16.97 1.95

Battalion made up of:
    Company COMPANY-42 (a Artillery company in Field emplaced posture) at 16.27 3.26
    Company COMPANY-43 (a Artillery company in Field emplaced posture) at 16.57 3.33
    Company COMPANY-41 (a Artillery company in Field emplaced posture) at 15.94 3.27

Battalion made up of:
    Company COMPANY-56 (a Artillery company in Field emplaced posture) at 19.42 1.81
    Company COMPANY-57 (a Artillery company in Field emplaced posture) at 19.99 1.95
    Company COMPANY-55 (a Artillery company in Field emplaced posture) at 19.14 1.76

Battalion made up of:
    Company COMPANY-59 (a Artillery company in Field emplaced posture) at 21.75 2.36
    Company COMPANY-60 (a Artillery company in Field emplaced posture) at 22.02 2.3
    Company COMPANY-58 (a Artillery company in Field emplaced posture) at 21.41 2.41

The following battalions were selected with belief 0.9 and conflict 0.0
Battalion made up of:
    Company COMPANY-61 (a SAM company in Field emplaced posture) at 22.32 6.51

The following battalions were selected with belief 0.9 and conflict 0.0
Battalion made up of:
    Company COMPANY-62 (a SAM company in Field emplaced posture) at 20.81 0.52

The following battalions were selected with belief 0.782128 and conflict 0.0
Battalion made up of:
    Company COMPANY-66 (a Engineer company in Assembly area posture) at 20.72 1.42
    Company COMPANY-65 (a Engineer company in Assembly area posture) at 20.35 1.23
    Company COMPANY-63 (a Engineer company in Assembly area posture) at 20.48 0.92
    Company COMPANY-64 (a Engineer company in Assembly area posture) at 20.99 1.12

The following battalions were selected with belief 0.9 and conflict 0.0
Battalion made up of:
    Company COMPANY-67 (a SAM company in Field emplaced posture) at 24.16 1.44

There may be map to image registration problems:
 DTED indicates company on impossible terrain for:
    Company COMPANY-55 (a Artillery company in Field emplaced posture) at 19.14 1.76
 Missing company to complete battalion for:
    Company COMPANY-46 (a Artillery company in Field emplaced posture) at 25.8 4.94
    Company COMPANY-47 (a Artillery company in Field emplaced posture) at 26.1 4.83
 Companies that do not form a battalion for:
    Company COMPANY-45 (a Artillery company in Field emplaced posture) at 26.92 2.21

There may be problems with misidentification of companies:
 Missing company to complete battalion for:
    Company COMPANY-46 (a Artillery company in Field emplaced posture) at 25.8 4.94
    Company COMPANY-47 (a Artillery company in Field emplaced posture) at 26.1 4.83
 Companies that do not form a battalion for:
    Company COMPANY-45 (a Artillery company in Field emplaced posture) at 26.92 2.21